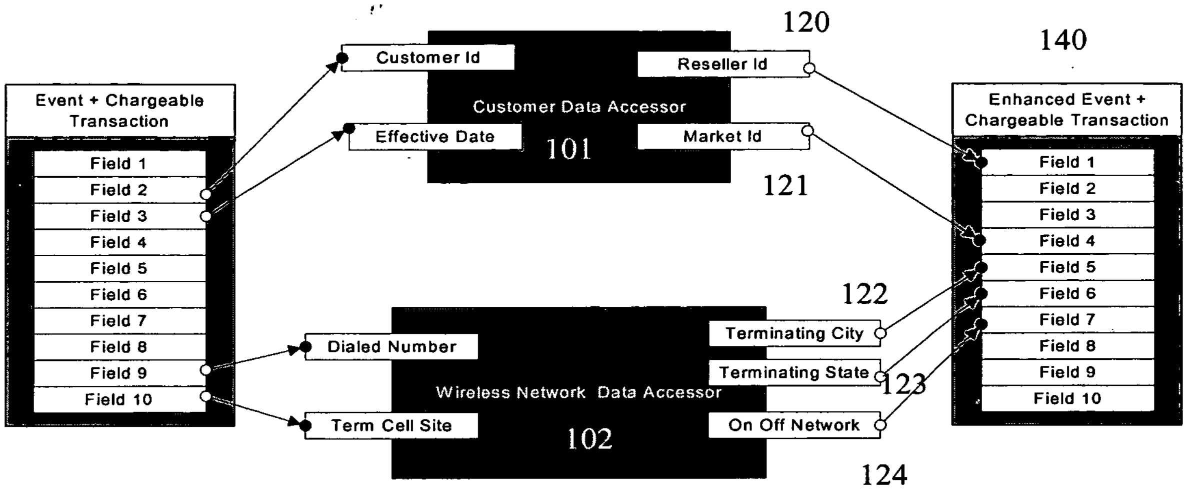

120
140
Customer Id
Reseller Id
Event + Chargeable Transaction
Customer Data Accessor
Enhanced Event + Chargeable Transaction
Effective Date
101
Market Id
121
Field 1
Field 2
Field 3
Field 4
Field 5
Field 6
Field 7
Field 8
Field 9
Field 10
122
Dialed Number
Terminating City
Wireless Network Data Accessor
Terminating State
123
Term Cell Site
102
On Off Network
124
Field 1
Field 2
Field 3
Field 4
Field 5
Field 6
Field 7
Field 8
Field 9
Field 10

Figure 1

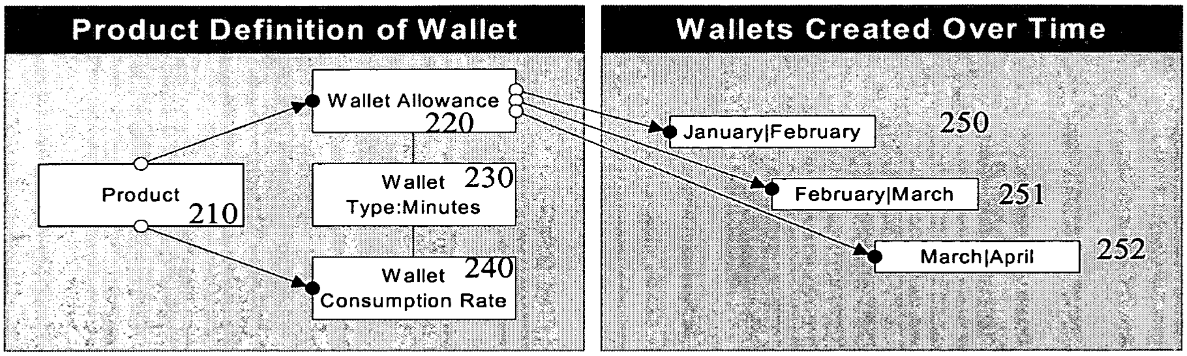

Product Definition of Wallet
Wallets Created Over Time
Wallet Allowance
220
Product
210
Wallet
230
Type:Minutes
Wallet
240
Consumption Rate
January|February
250
February|March
251
March|April
252

Figure 2

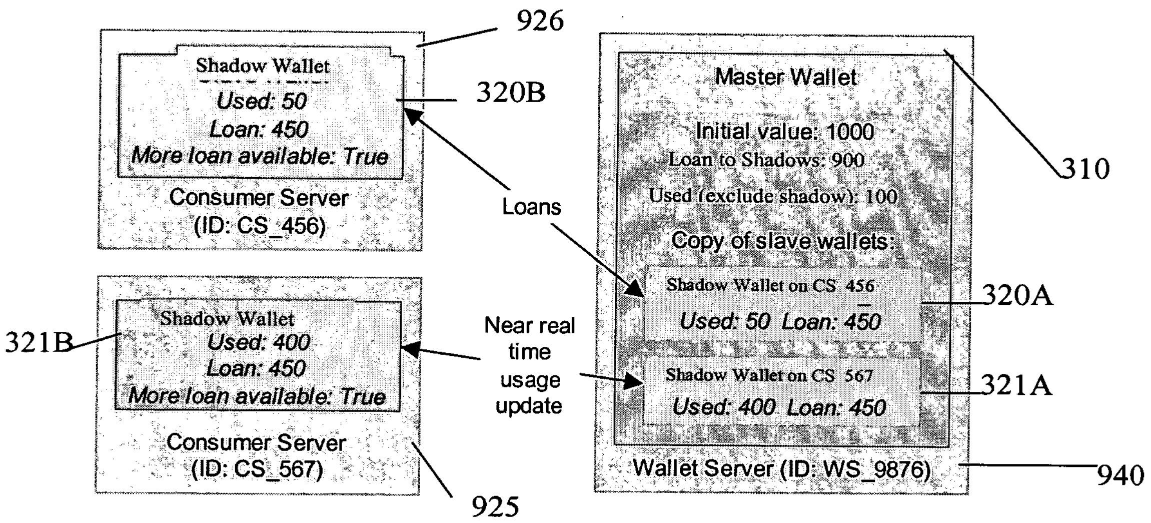

926
Shadow Wallet
Used: 50
Loan: 450
More loan available: True
320B
Consumer Server
(ID: CS_456)
Loans
Master Wallet
Initial value: 1000
Loan to Shadows: 900
Used (exclude shadow): 100
310
Copy of slave wallets:
Shadow Wallet on CS_456
Used: 50 Loan: 450
320A
321B
Shadow Wallet
Used: 400
Loan: 450
More loan available: True
Near real time usage update
Shadow Wallet on CS_567
Used: 400 Loan: 450
321A
Consumer Server
(ID: CS_567)
Wallet Server (ID: WS_9876)
940
925

Figure 3

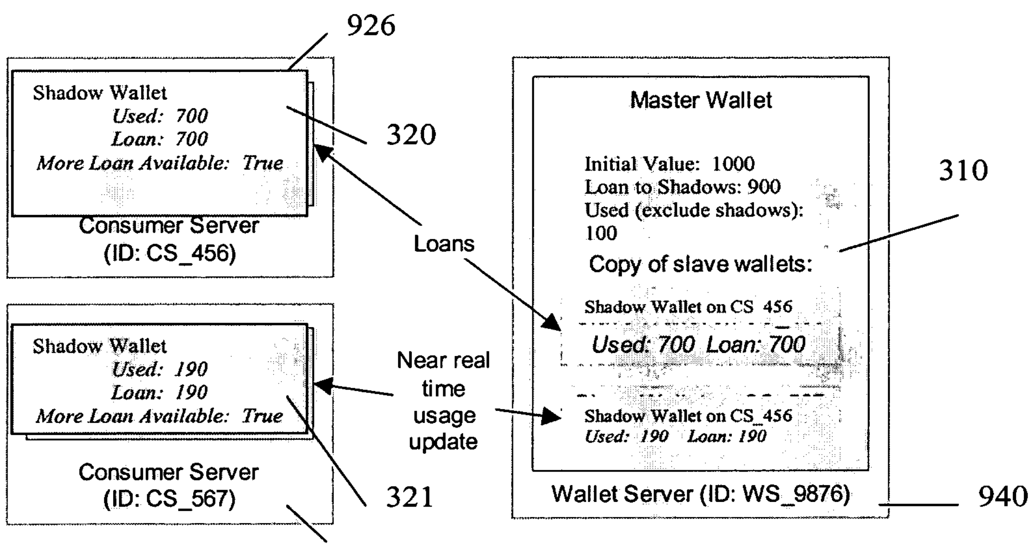

926
Shadow Wallet
Used: 700
Loan: 700
More Loan Available: True
Consumer Server
(ID: CS_456)
320
Loans
Master Wallet
Initial Value: 1000
Loan to Shadows: 900
Used (exclude shadows): 100
Copy of slave wallets:
Shadow Wallet on CS 456
Used: 700 Loan: 700
310
Shadow Wallet
Used: 190
Loan: 190
More Loan Available: True
Near real time usage update
Shadow Wallet on CS_456
Used: 190 Loan: 190
Consumer Server
(ID: CS_567)
321
Wallet Server (ID: WS_9876)
940
925

Figure 4

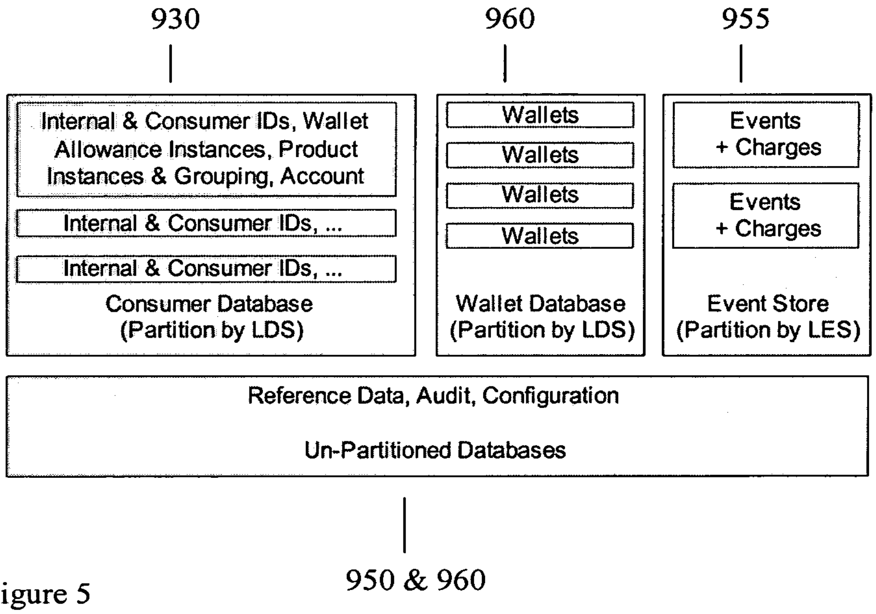

930
960
955
Internal & Consumer IDs, Wallet Allowance Instances, Product Instances & Grouping, Account
Internal & Consumer IDs, ...
Internal & Consumer IDs, ...
Consumer Database (Partition by LDS)
Wallets
Wallets
Wallets
Wallets
Wallet Database (Partition by LDS)
Events + Charges
Events + Charges
Event Store (Partition by LES)
Reference Data, Audit, Configuration
Un-Partitioned Databases
950 & 960

Figure 5

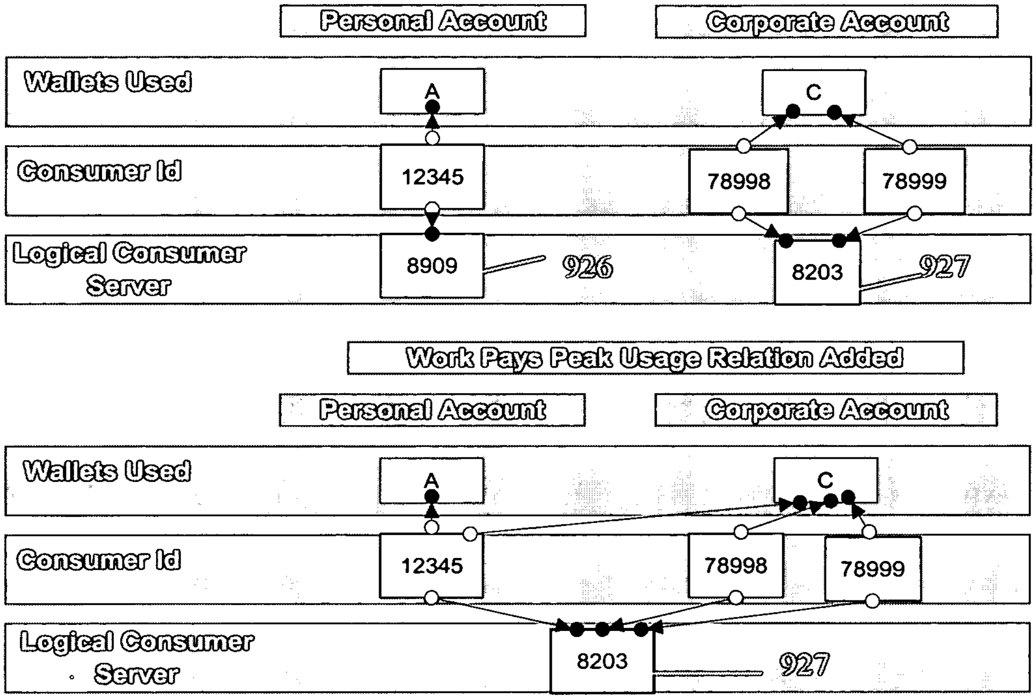

Personal Account
Corporate Account
Wallets Used
A
C
Consumer Id
12345
78998
78999
Logical Consumer Server
8909
926
8203
927
Work Pays Peak Usage Relation Added
Personal Account
Corporate Account
Wallets Used
A
C
Consumer Id
12345
78998
78999
Logical Consumer Server
8203
927

Figure 6

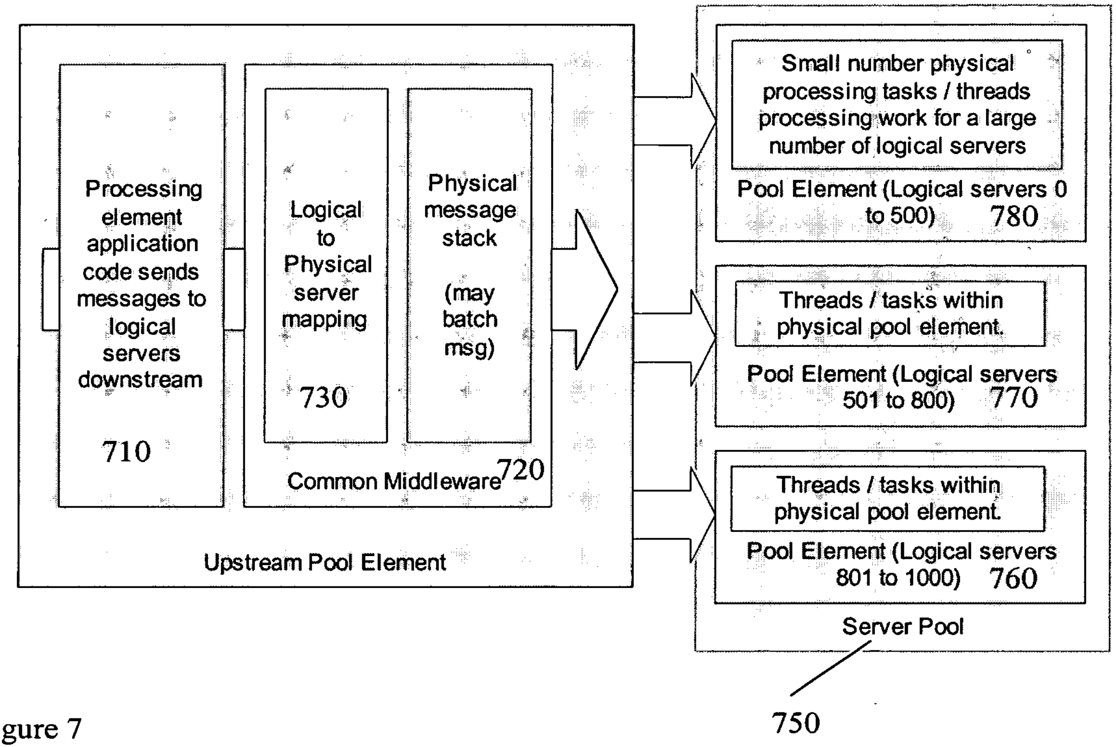

Processing element application code sends messages to logical servers downstream
710
Logical to Physical server mapping
730
Physical message stack
(may batch msg)
Common Middleware
720
Upstream Pool Element
Small number physical processing tasks / threads processing work for a large number of logical servers
Pool Element (Logical servers 0 to 500)
780
Threads / tasks within physical pool element.
Pool Element (Logical servers 501 to 800)
770
Threads / tasks within physical pool element.
Pool Element (Logical servers 801 to 1000)
760
Server Pool
750

Figure 7

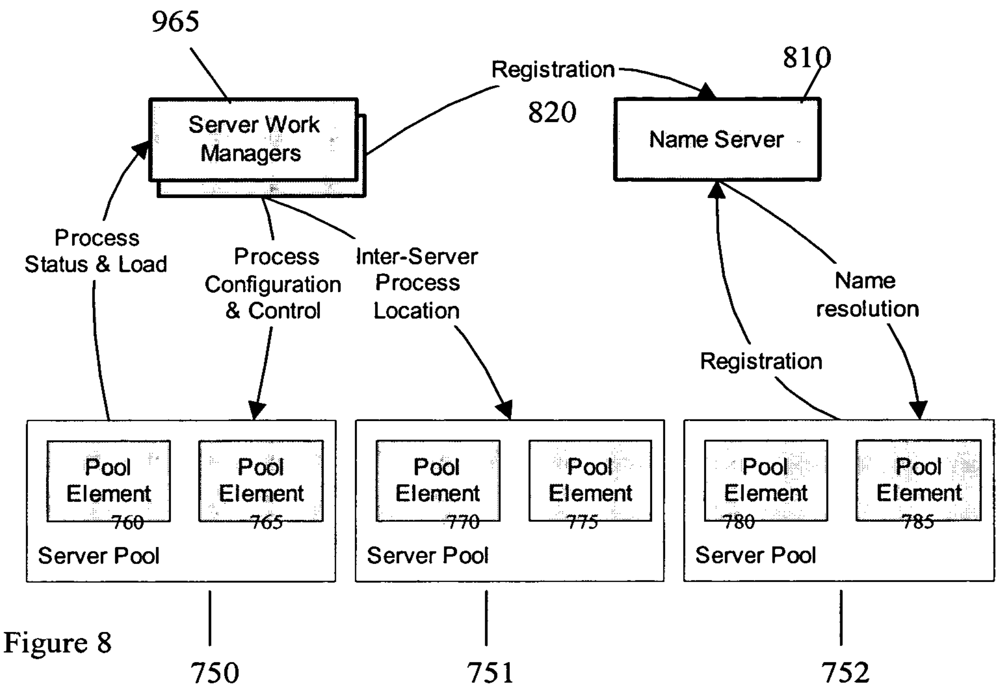

965
Server Work Managers
Registration
820
810
Name Server
Process Status & Load
Process Configuration & Control
Inter-Server Process Location
Name resolution
Registration
Pool Element 760
Pool Element 765
Server Pool
Pool Element 770
Pool Element 775
Server Pool
Pool Element 780
Pool Element 785
Server Pool
750
751
752

Figure 8

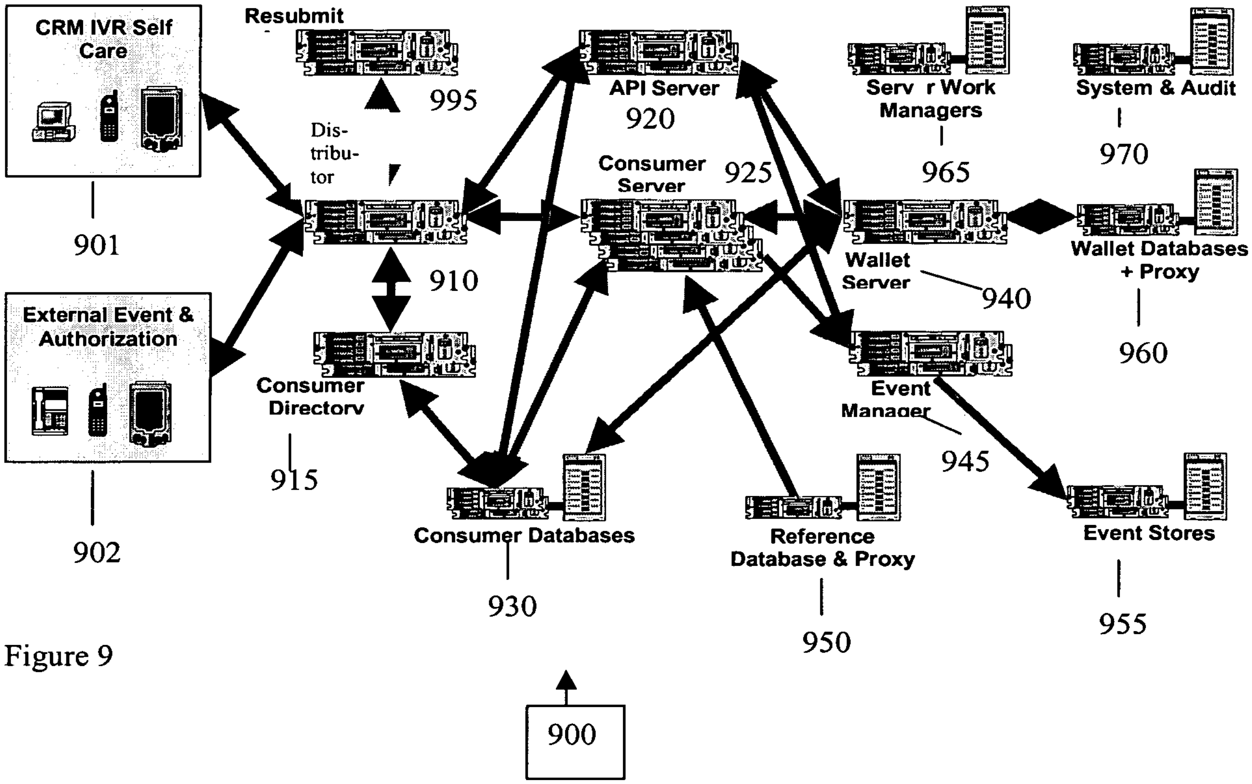
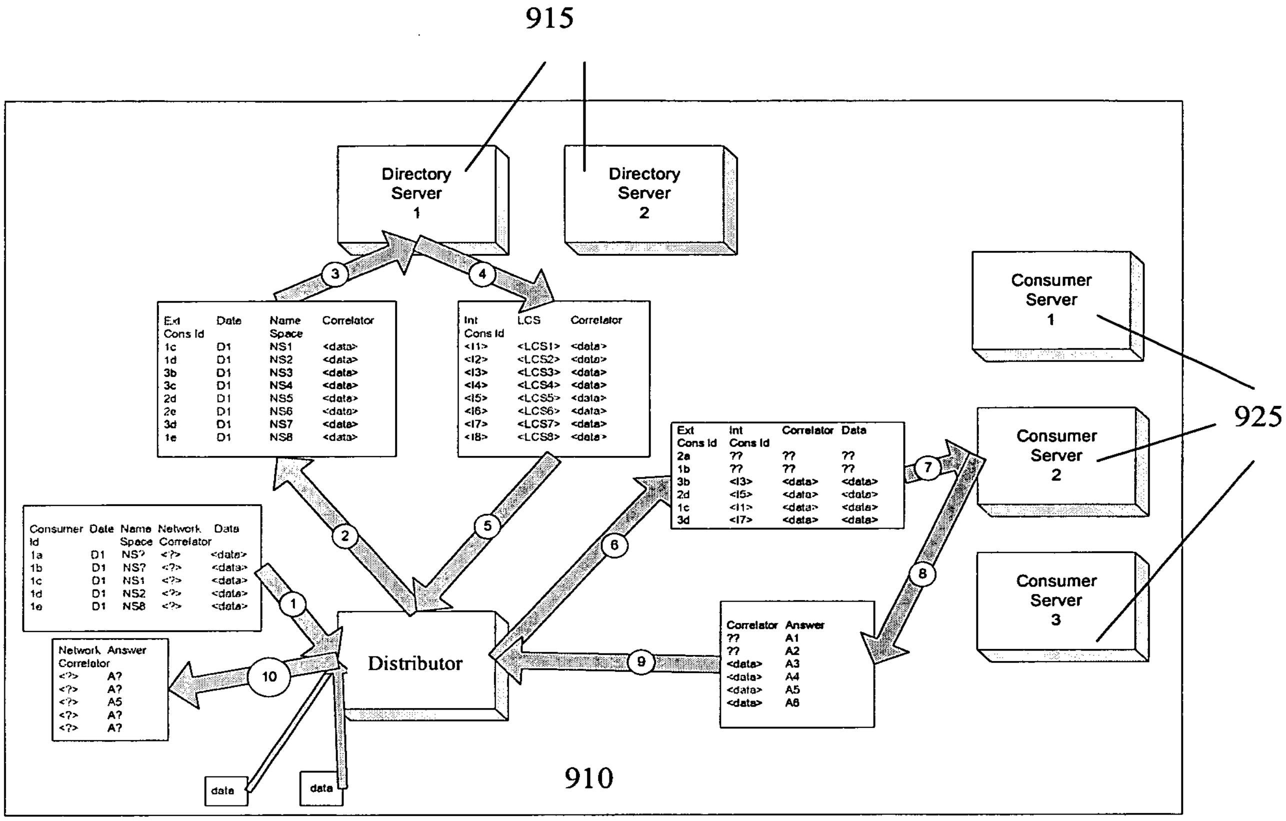

CRM IVR Self Care
901
External Event & Authorization
902
Resubmit
995
Dis-tribu-tor
910
Consumer Directory
915
API Server
920
Consumer Server
925
Consumer Databases
930
Reference Database & Proxy
950
Serv r Work Managers
965
System & Audit
970
Wallet Server
940
Wallet Databases + Proxy
960
Event Manager
945
Event Stores
955
900

Figure 9



Consumer Id	Date	Name Space	Network Correlator	Data
1a	D1	NS?	<?>	<data>
1b	D1	NS?	<?>	<data>
1c	D1	NS1	<?>	<data>
1d	D1	NS2	<?>	<data>
1e	D1	NS8	<?>	<data>

Ext Cons Id	Date	Name Space	Correlator
1c	D1	NS1	<data>
1d	D1	NS2	<data>
3b	D1	NS3	<data>
3c	D1	NS4	<data>
2d	D1	NS5	<data>
2e	D1	NS6	<data>
3d	D1	NS7	<data>
1e	D1	NS8	<data>

Int Cons Id	LCS	Correlator
<I1>	<LCS1>	<data>
<I2>	<LCS2>	<data>
<I3>	<LCS3>	<data>
<I4>	<LCS4>	<data>
<I5>	<LCS5>	<data>
<I6>	<LCS6>	<data>
<I7>	<LCS7>	<data>
<I8>	<LCS8>	<data>

Ext Cons Id	Int Cons Id	Correlator	Data
2a	??	??	??
1b	??	??	??
3b	<I3>	<data>	<data>
2d	<I5>	<data>	<data>
1c	<I1>	<data>	<data>
3d	<I7>	<data>	<data>

Correlator	Answer
??	A1
??	A2
<data>	A3
<data>	A4
<data>	A5
<data>	A6

Network Correlator	Answer
<?>	A?
<?>	A?
<?>	A5
<?>	A?
<?>	A?

Figure 10

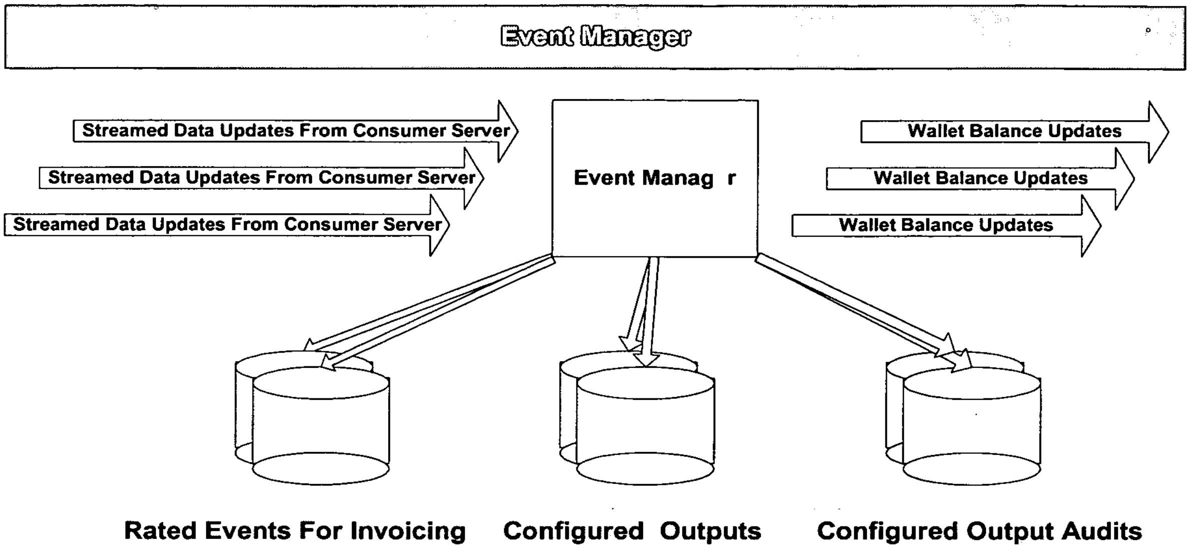

Event Manager
Streamed Data Updates From Consumer Server
Streamed Data Updates From Consumer Server
Streamed Data Updates From Consumer Server
Event Manag r
Wallet Balance Updates
Wallet Balance Updates
Wallet Balance Updates
Rated Events For Invoicing
Configured Outputs
Configured Output Audits

Figure 11

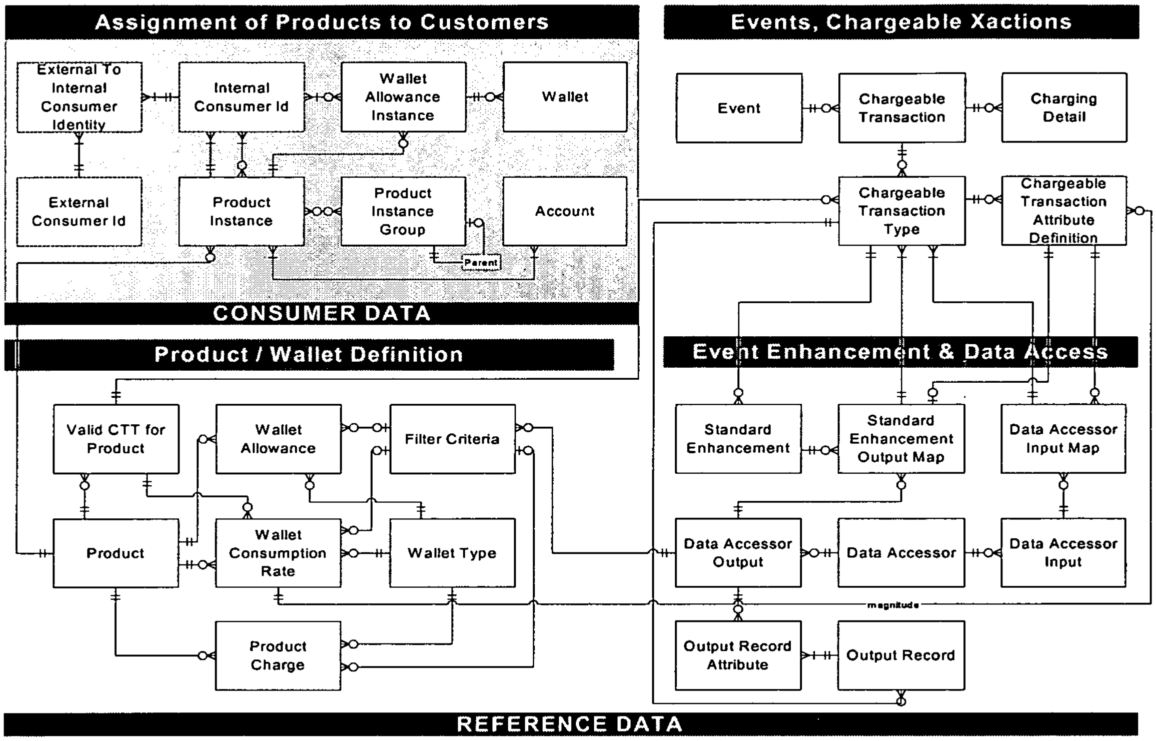

Assignment of Products to Customers
External To Internal Consumer Identity
Internal Consumer Id
Wallet Allowance Instance
Wallet
External Consumer Id
Product Instance
Product Instance Group
Account
Parent
CONSUMER DATA
Events, Chargeable Xactions
Event
Chargeable Transaction
Charging Detail
Chargeable Transaction Type
Chargeable Transaction Attribute Definition
Product / Wallet Definition
Valid CTT for Product
Wallet Allowance
Filter Criteria
Product
Wallet Consumption Rate
Wallet Type
Product Charge
Event Enhancement & Data Access
Standard Enhancement
Standard Enhancement Output Map
Data Accessor Input Map
Data Accessor Output
Data Accessor
Data Accessor Input
magnitude
Output Record Attribute
Output Record
REFERENCE DATA

Figure 12

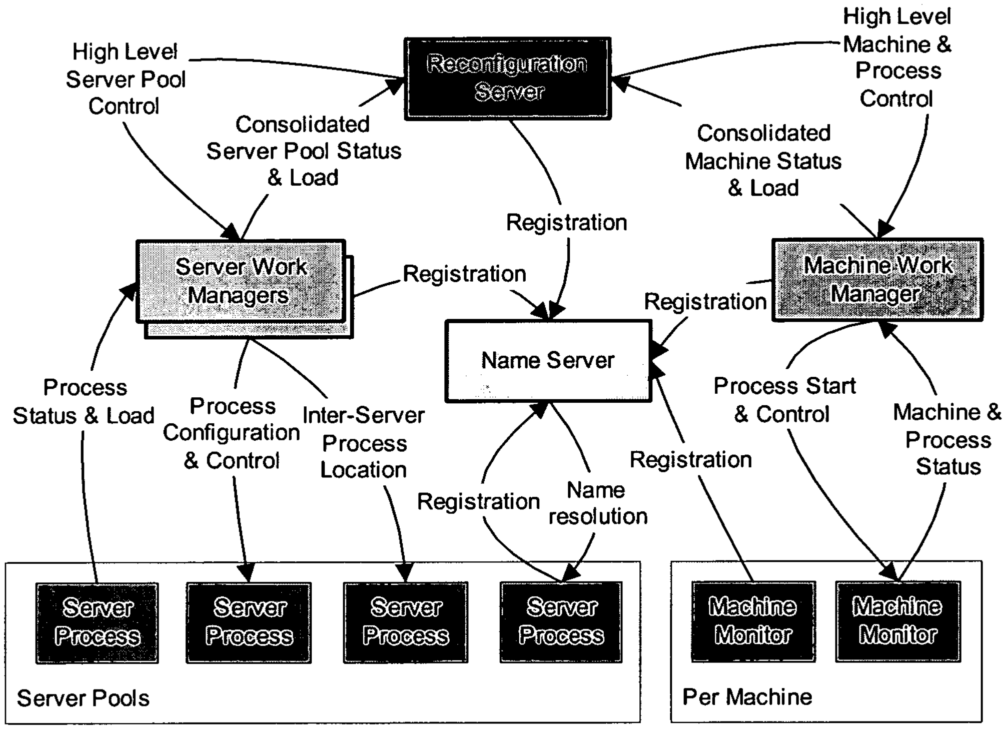

High Level Server Pool Control
Reconfiguration Server
High Level Machine & Process Control
Consolidated Server Pool Status & Load
Consolidated Machine Status & Load
Registration
Server Work Managers
Registration
Registration
Machine Work Manager
Name Server
Process Status & Load
Process Configuration & Control
Inter-Server Process Location
Process Start & Control
Machine & Process Status
Registration
Registration
Name resolution
Server Process
Server Process
Server Process
Server Process
Machine Monitor
Machine Monitor
Server Pools
Per Machine

Figure 13

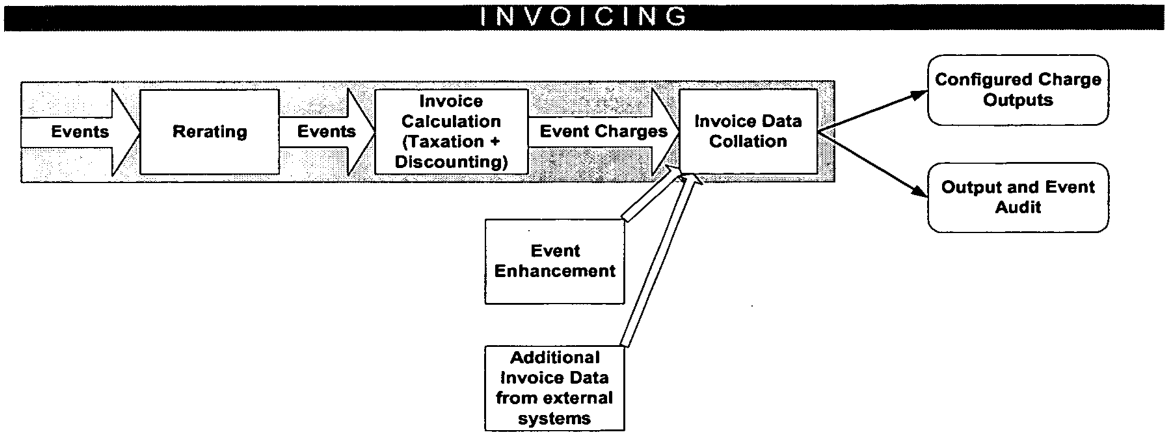

INVOICING
Events
Rerating
Events
Invoice Calculation (Taxation + Discounting)
Event Charges
Invoice Data Collation
Configured Charge Outputs
Output and Event Audit
Event Enhancement
Additional Invoice Data from external systems

Figure 14

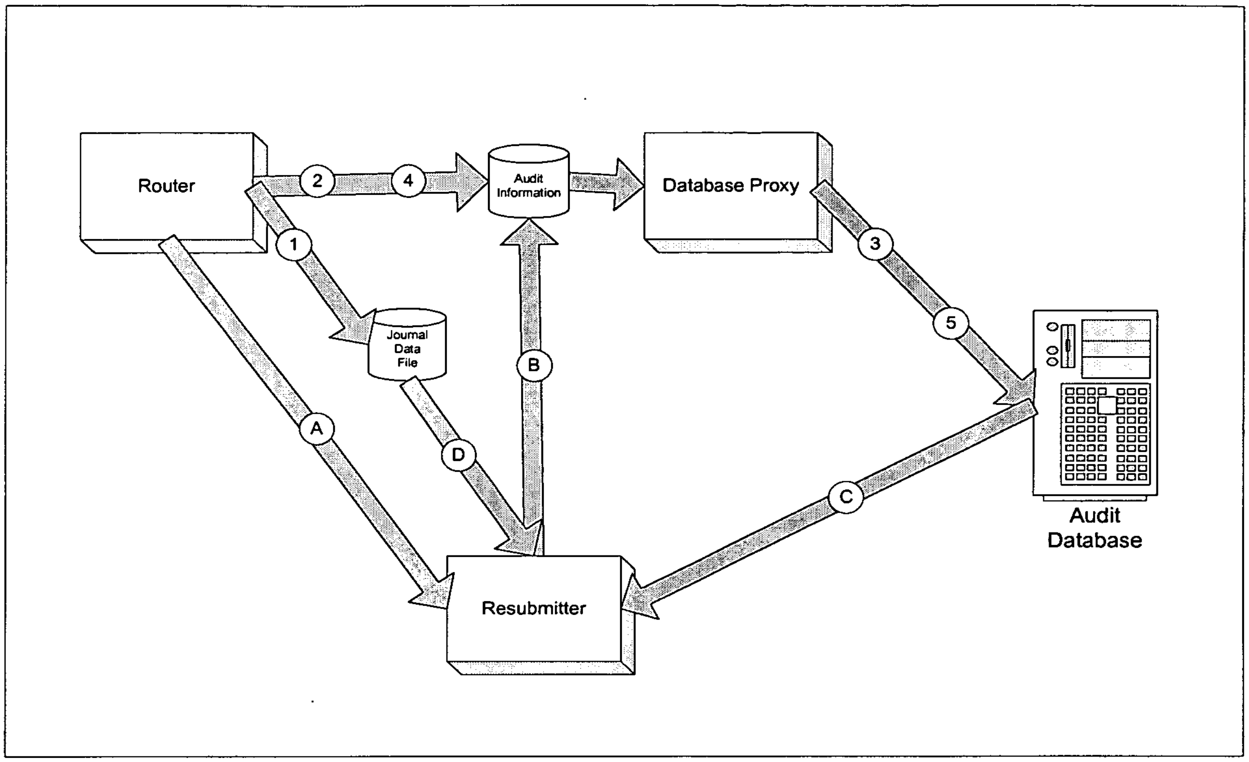
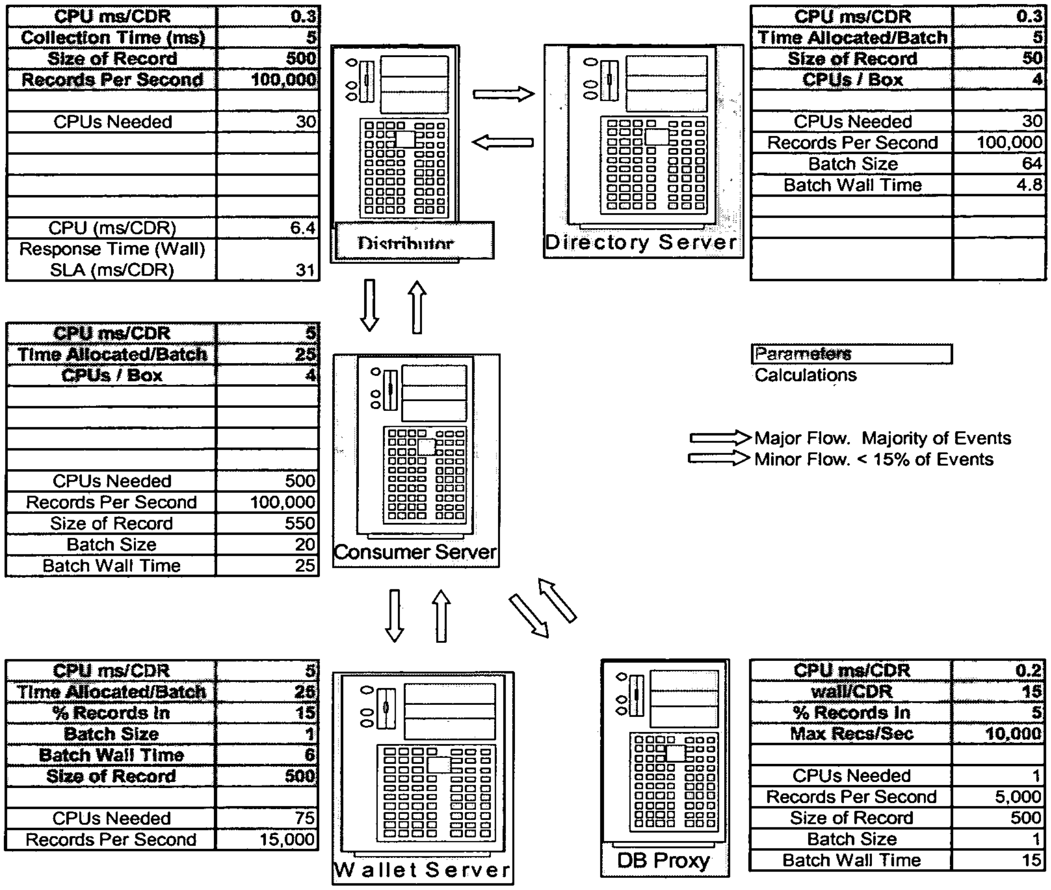

Router
2
4
Audit Information
Database Proxy
1
3
Journal Data File
B
5
A
D
C
Audit Database
Resubmitter

Figure 15



CPU ms/CDR	**0.3**
Collection Time (ms)	**5**
Size of Record	**500**
Records Per Second	**100,000**
CPUs Needed	30
CPU (ms/CDR)	6.4
Response Time (Wall) SLA (ms/CDR)	31

CPU ms/CDR	**0.3**
Time Allocated/Batch	**5**
Size of Record	**50**
CPUs / Box	**4**
CPUs Needed	30
Records Per Second	100,000
Batch Size	64
Batch Wall Time	4.8

CPU ms/CDR	**5**
Time Allocated/Batch	**25**
CPUs / Box	**4**
CPUs Needed	500
Records Per Second	100,000
Size of Record	550
Batch Size	20
Batch Wall Time	25

Parameters

Calculations

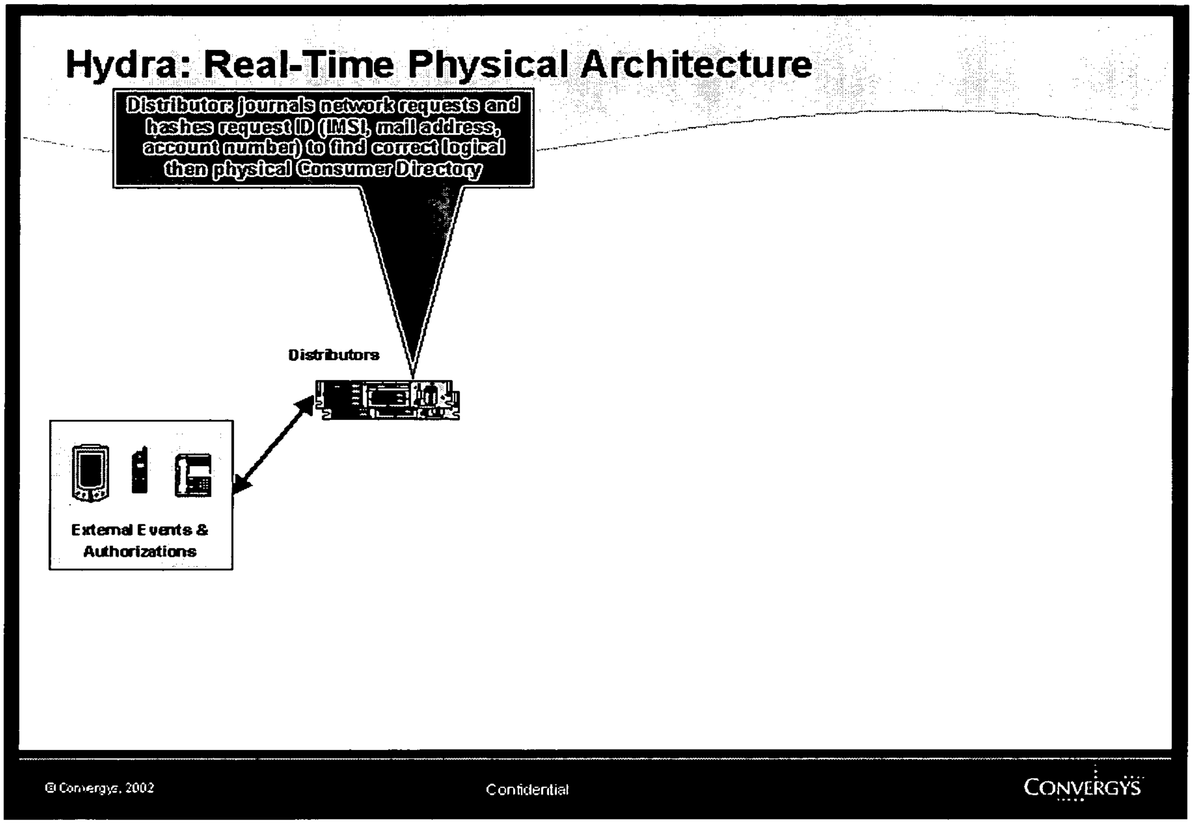
⟹ Major Flow. Majority of Events

⟹ Minor Flow. < 15% of Events

CPU ms/CDR	**5**
Time Allocated/Batch	**25**
% Records In	**15**
Batch Size	**1**
Batch Wall Time	**6**
Size of Record	**500**
CPUs Needed	75
Records Per Second	15,000

CPU ms/CDR	**0.2**
wall/CDR	**15**
% Records In	**5**
Max Recs/Sec	**10,000**
CPUs Needed	1
Records Per Second	5,000
Size of Record	500
Batch Size	1
Batch Wall Time	15

Figure 16

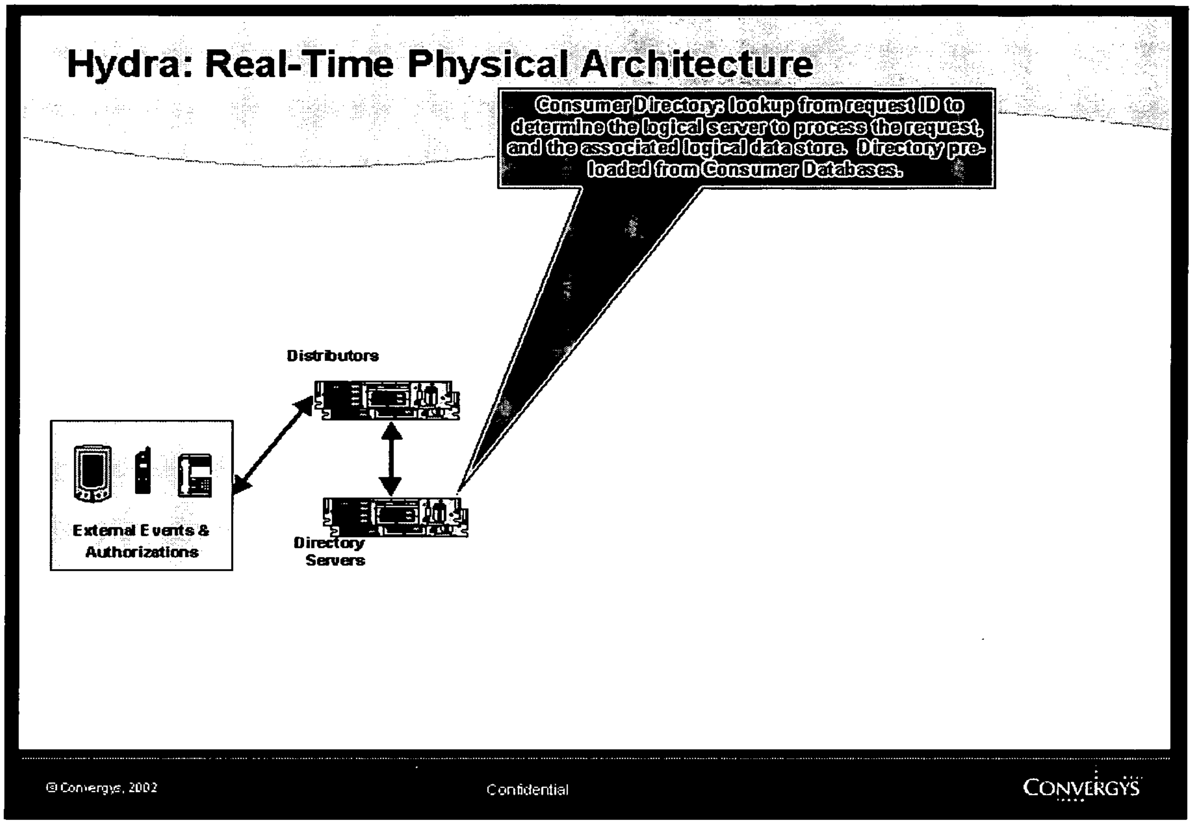


Hydra: Real-Time Physical Architecture

Distributor: journals network requests and hashes request ID (IMSI, mail address, account number) to find correct logical then physical Consumer Directory

© Convergys, 2002 Confidential CONVERGYS

Figure 17



Hydra: Real-Time Physical Architecture

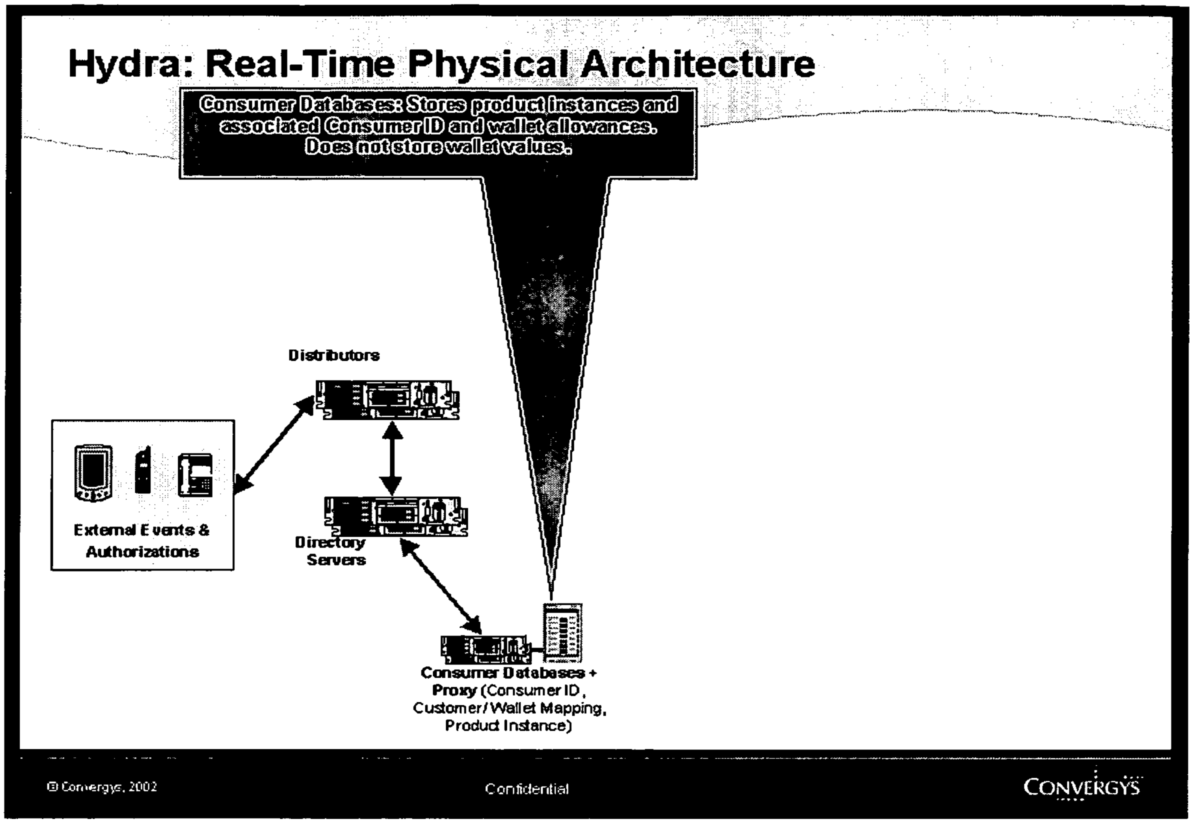
Consumer Directory: lookup from request ID to determine the logical server to process the request, and the associated logical data store. Directory pre-loaded from Consumer Databases.

© Convergys, 2002

Confidential

CONVERGYS

Figure 18



Hydra: Real-Time Physical Architecture

Consumer Databases: Stores product instances and associated Consumer ID and wallet allowances. Does not store wallet values.

Distributors

External Events & Authorizations

Directory Servers

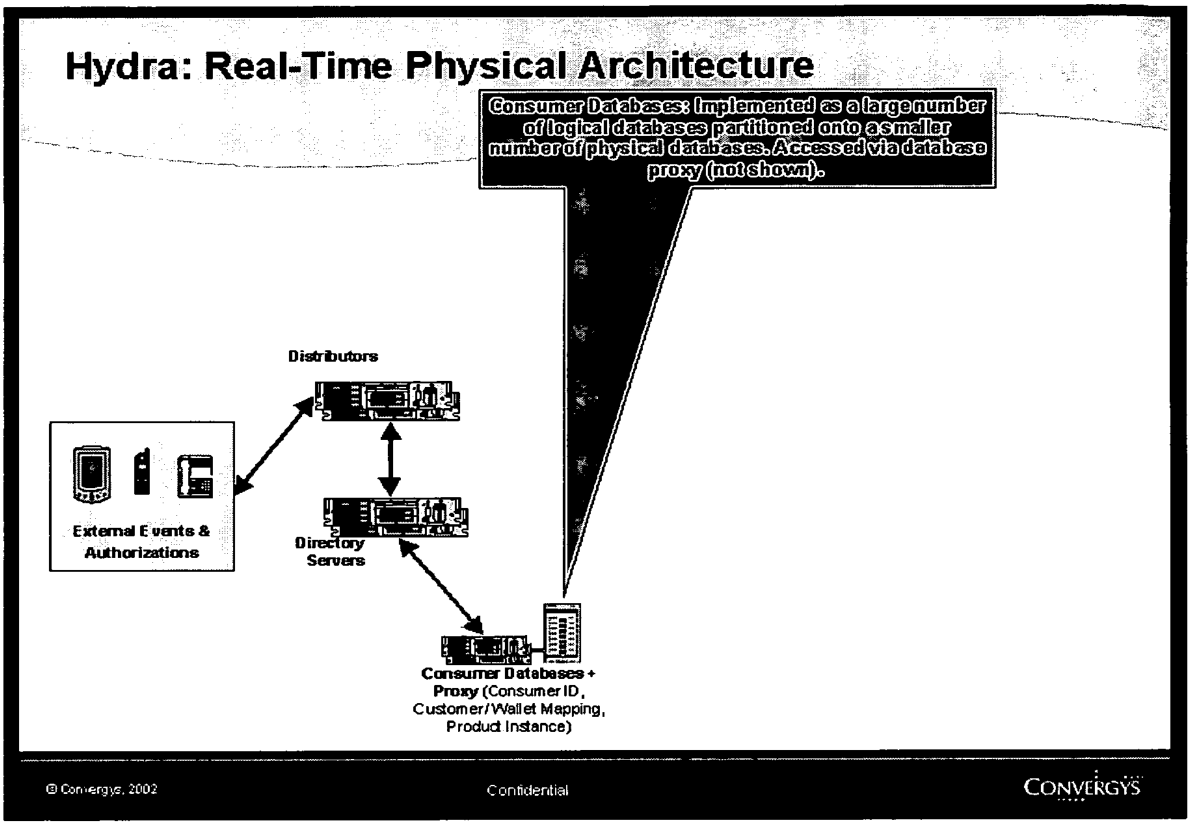
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

© Convergys, 2002

Confidential

CONVERGYS

Figure 19



Hydra: Real-Time Physical Architecture

Consumer Databases: Implemented as a large number of logical databases partitioned onto a smaller number of physical databases. Accessed via database proxy (not shown).

Distributors

External Events & Authorizations

Directory Servers

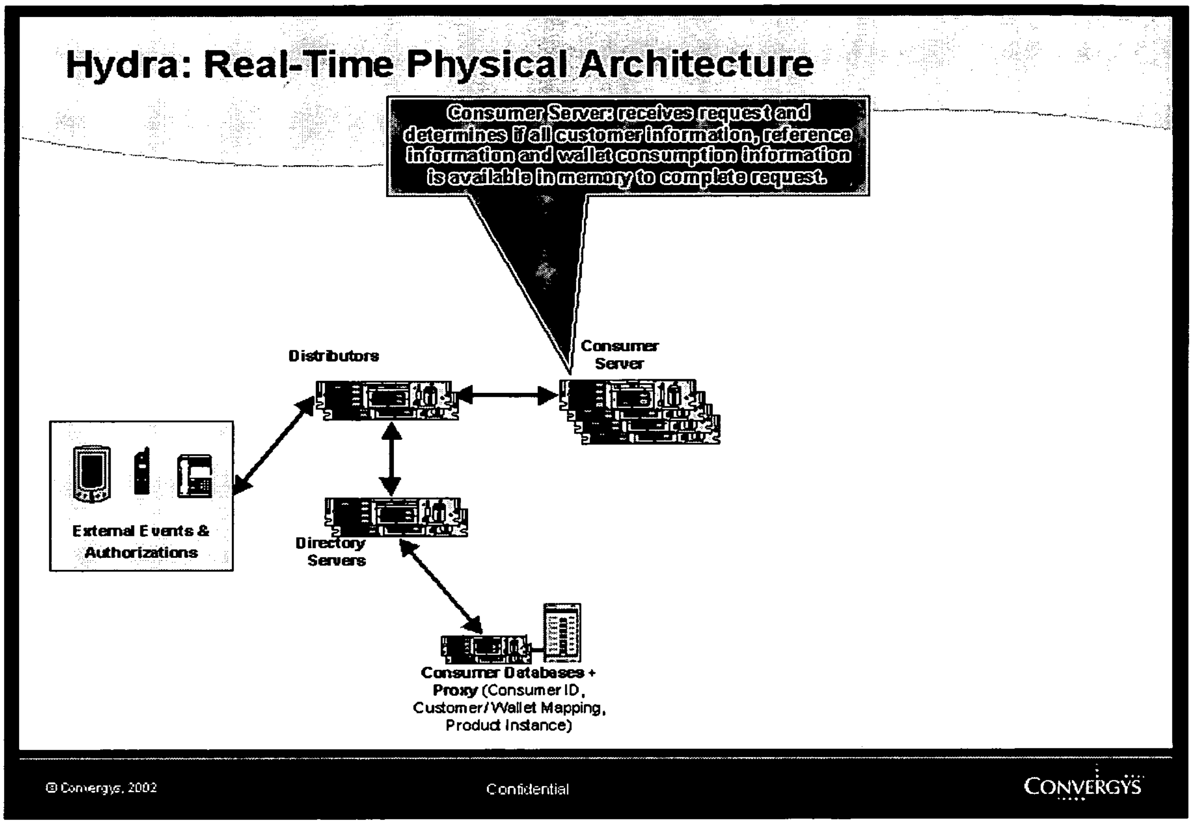
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

© Convergys, 2002

Confidential

CONVERGYS

Figure 20



Hydra: Real-Time Physical Architecture

Consumer Server: receives request and determines if all customer information, reference information and wallet consumption information is available in memory to complete request.

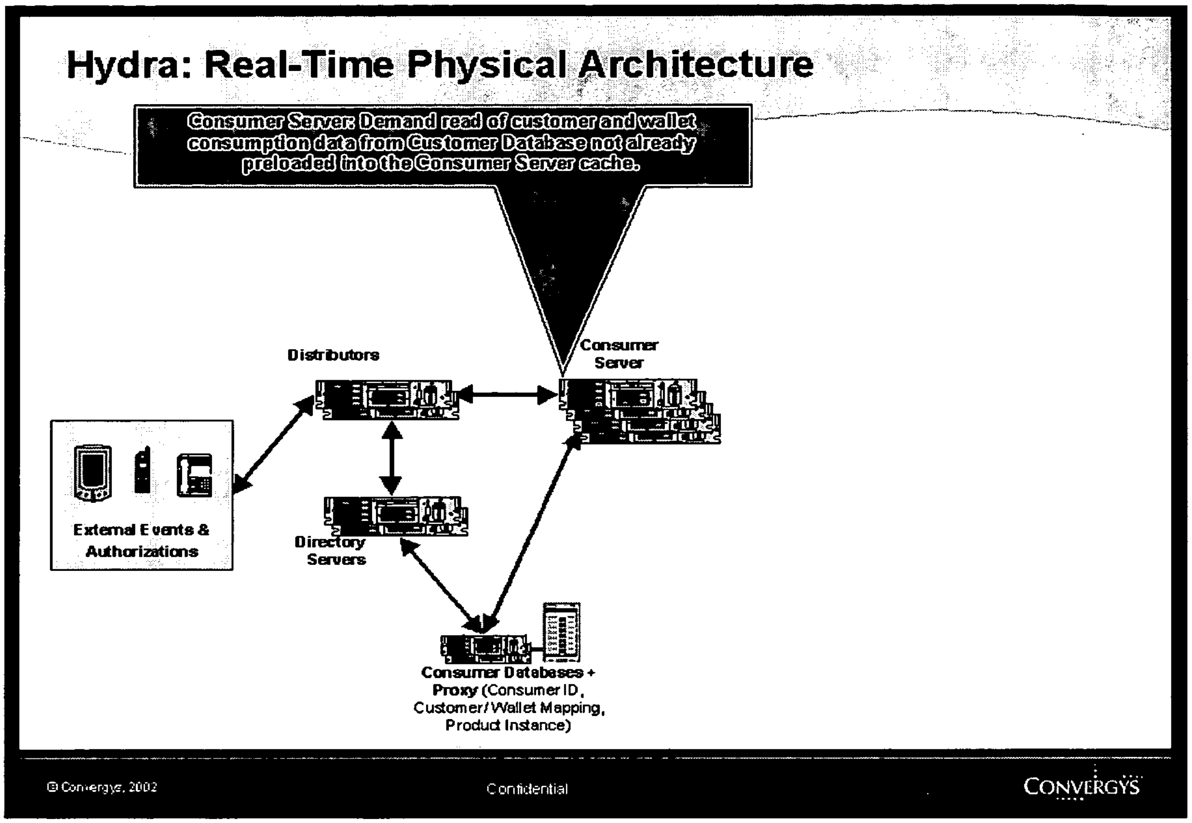
Distributors
Consumer Server
External Events & Authorizations
Directory Servers
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

© Convergys, 2002

Confidential

CONVERGYS

Figure 21



Hydra: Real-Time Physical Architecture

Consumer Server: Demand read of customer and wallet consumption data from Customer Database not already preloaded into the Consumer Server cache.

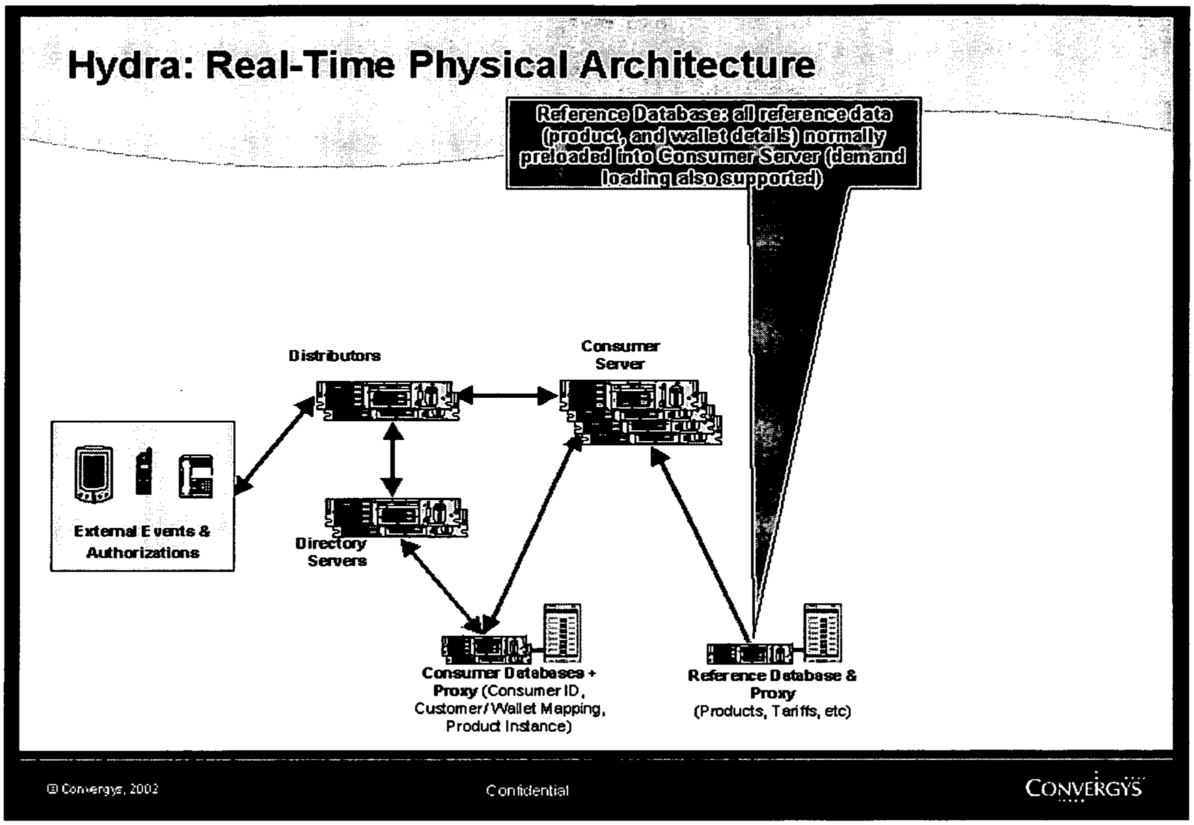
Distributors
Consumer Server
External Events & Authorizations
Directory Servers
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

© Convergys, 2002

Confidential

CONVERGYS

Figure 22



Hydra: Real-Time Physical Architecture

Reference Database: all reference data (product, and wallet details) normally preloaded into Consumer Server (demand loading also supported)

Distributors

Consumer Server

External Events & Authorizations

Directory Servers

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

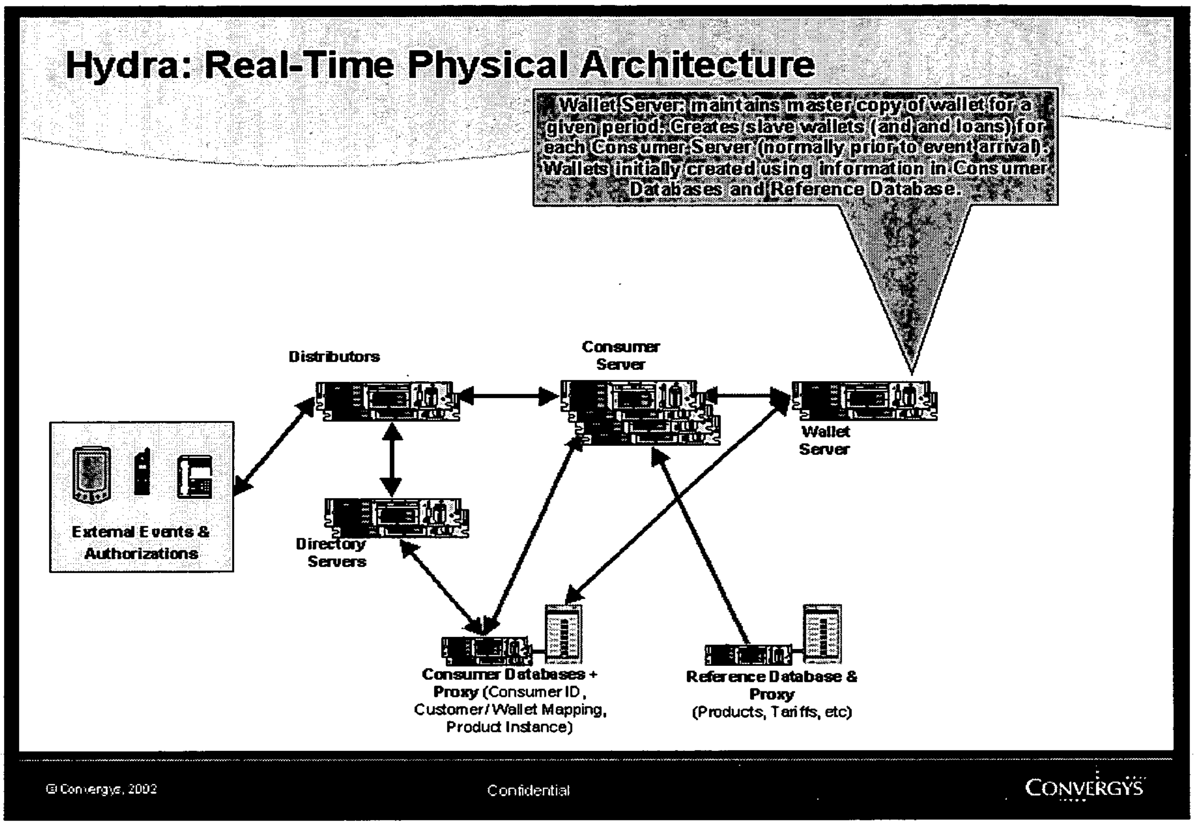
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 23



Hydra: Real-Time Physical Architecture

Wallet Server: maintains master copy of wallet for a given period. Creates slave wallets (and and loans) for each Consumer Server (normally prior to event arrival). Wallets initially created using information in Consumer Databases and Reference Database.

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

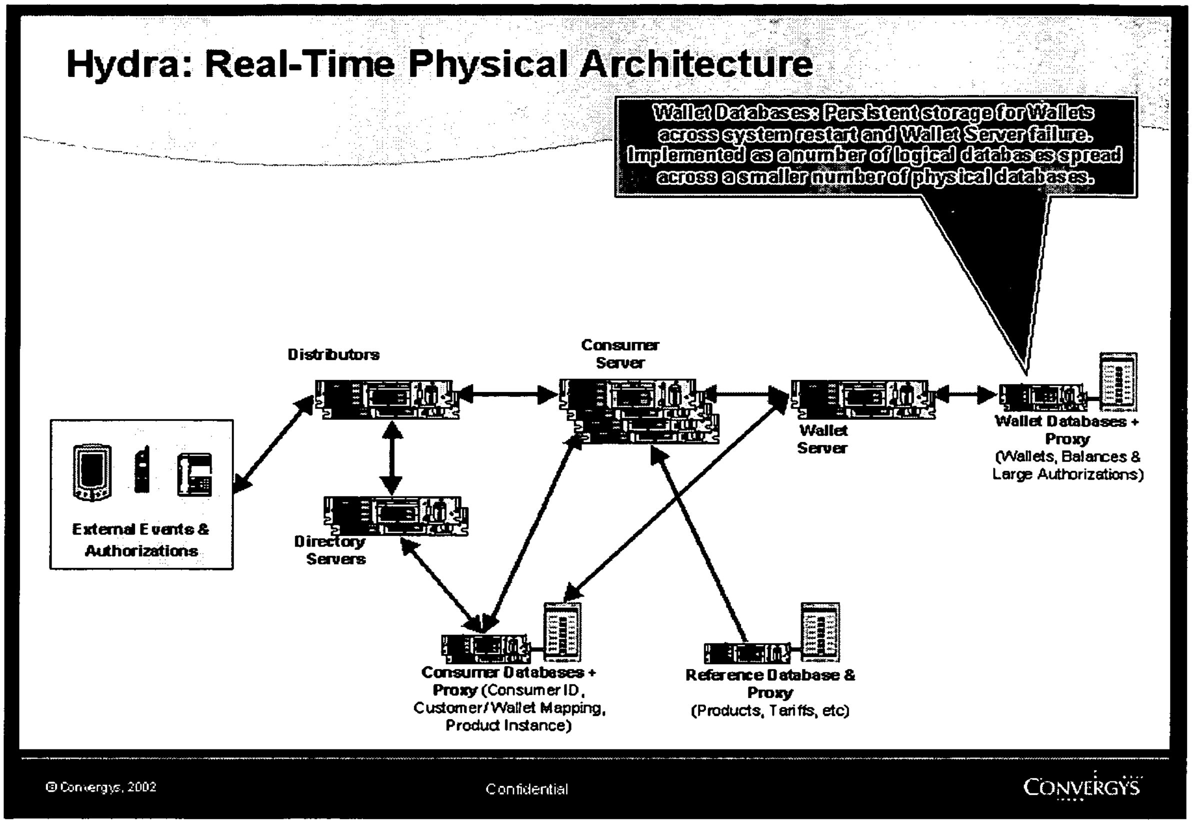
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 24



Hydra: Real-Time Physical Architecture

Wallet Databases: Persistent storage for Wallets across system restart and Wallet Server failure. Implemented as a number of logical databases spread across a smaller number of physical databases.

Distributors

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

External Events & Authorizations

Directory Servers

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

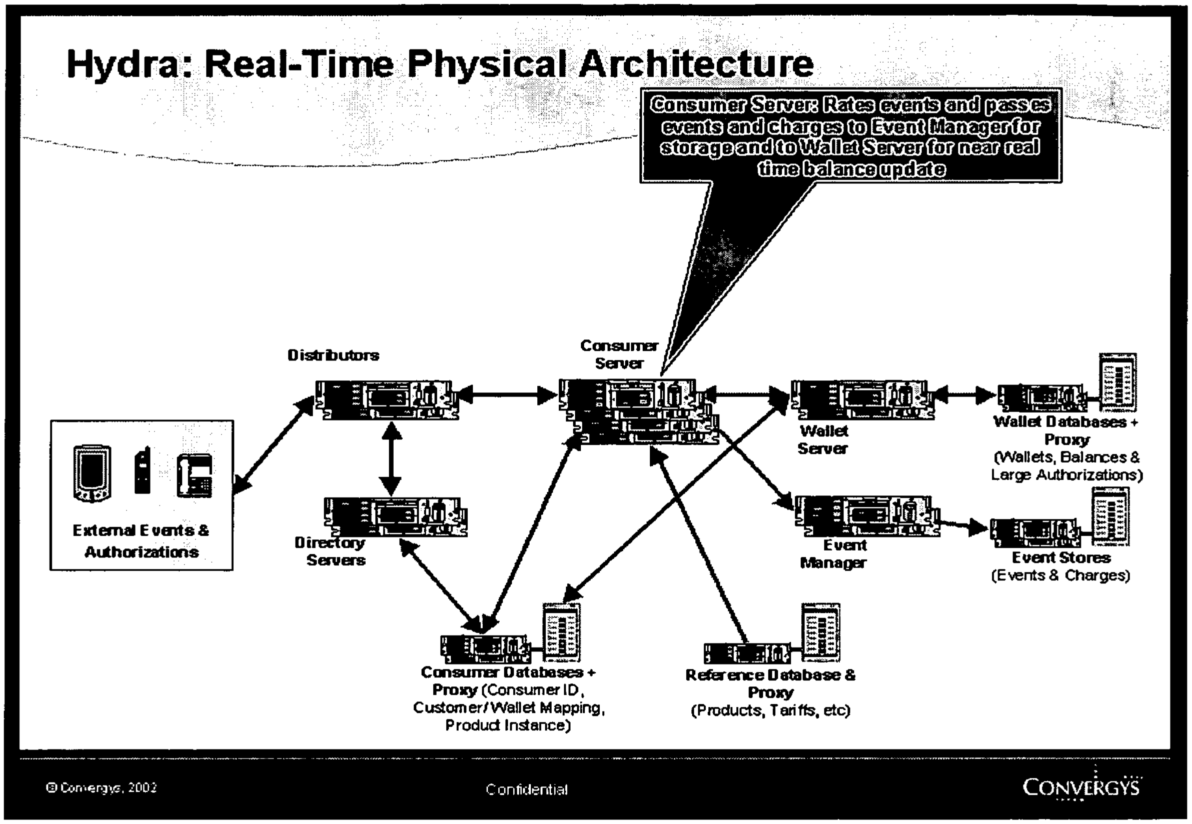
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 25



Hydra: Real-Time Physical Architecture

Consumer Server: Rates events and passes events and charges to Event Manager for storage and to Wallet Server for near real time balance update

External Events & Authorizations

Distributors

Directory Servers

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

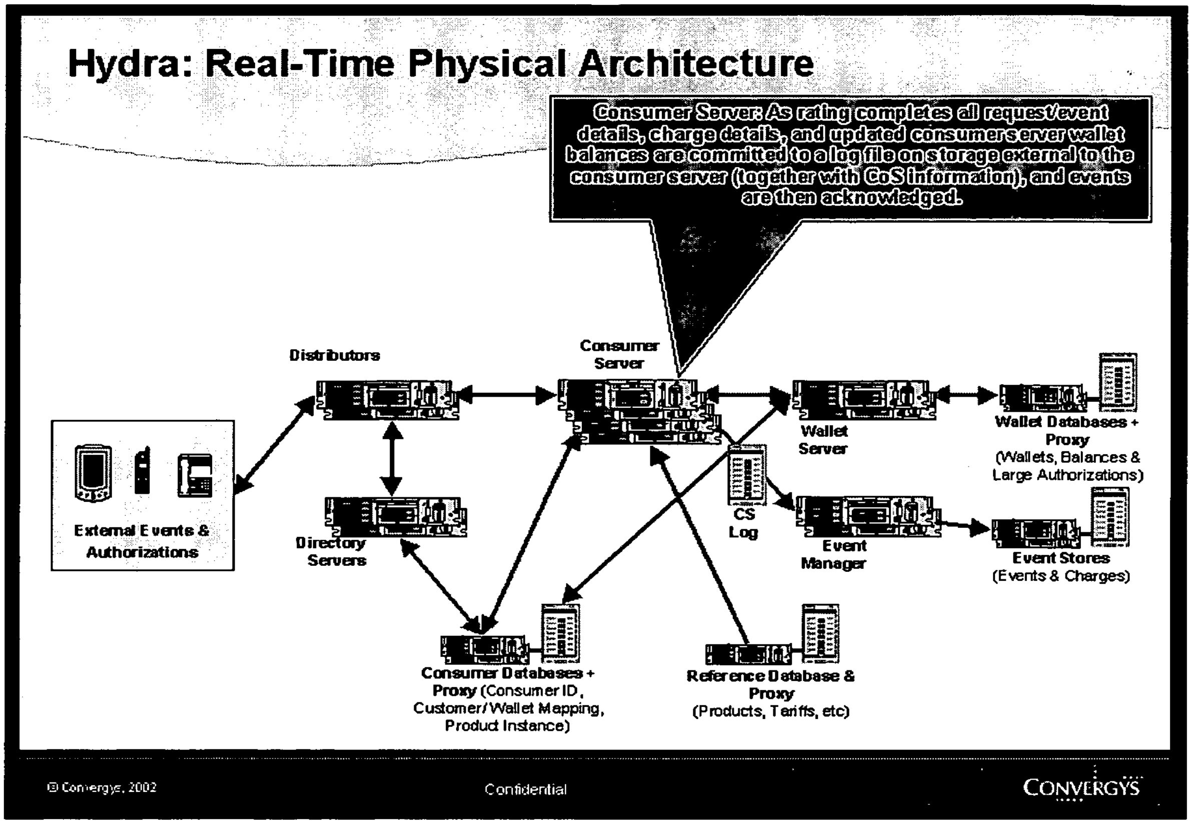
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 26



Hydra: Real-Time Physical Architecture

Consumer Server: As rating completes all request/event details, charge details, and updated consumerserver wallet balances are committed to a log file on storage external to the consumer server (together with CoS information), and events are then acknowledged.

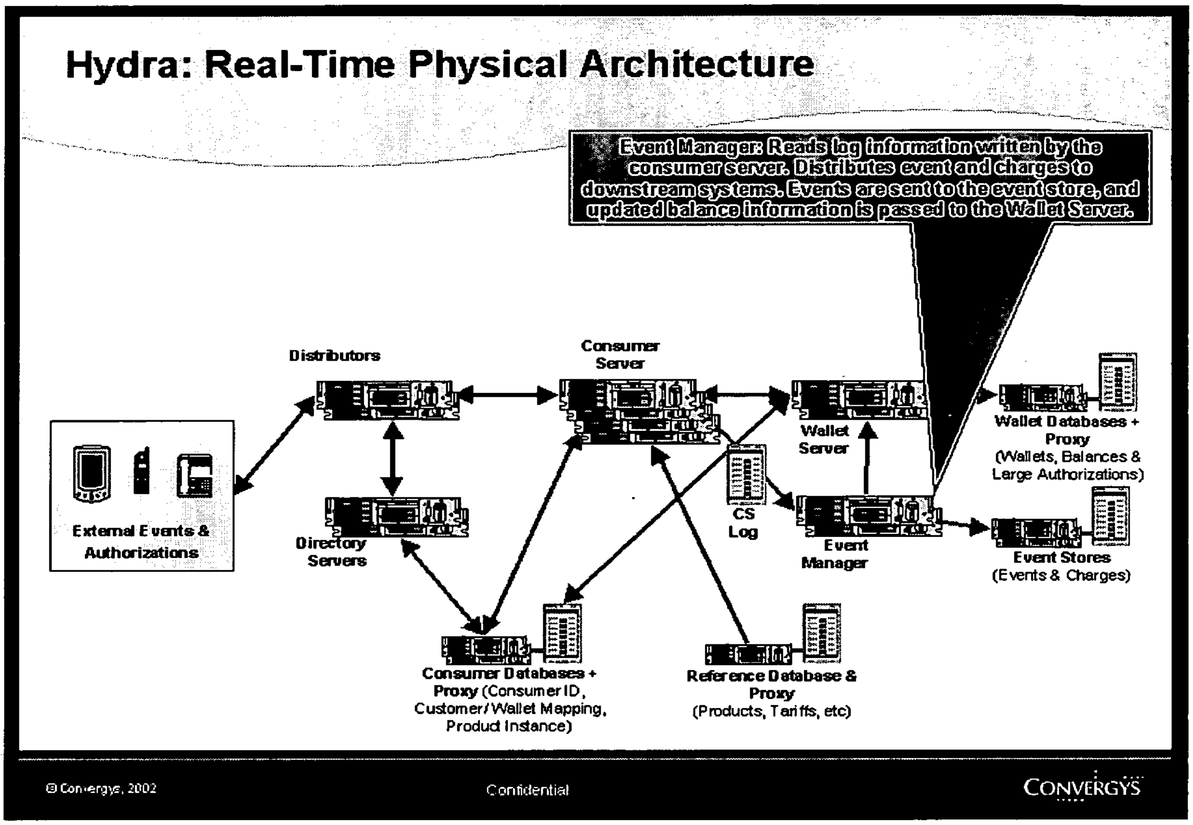
Distributors
Consumer Server
Wallet Server
Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)
CS Log
External Events & Authorizations
Directory Servers
Event Manager
Event Stores (Events & Charges)
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 27



Hydra: Real-Time Physical Architecture

Event Manager: Reads log information written by the consumer server. Distributes event and charges to downstream systems. Events are sent to the event store, and updated balance information is passed to the Wallet Server.

Distributors

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

External Events & Authorizations

Directory Servers

CS Log

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

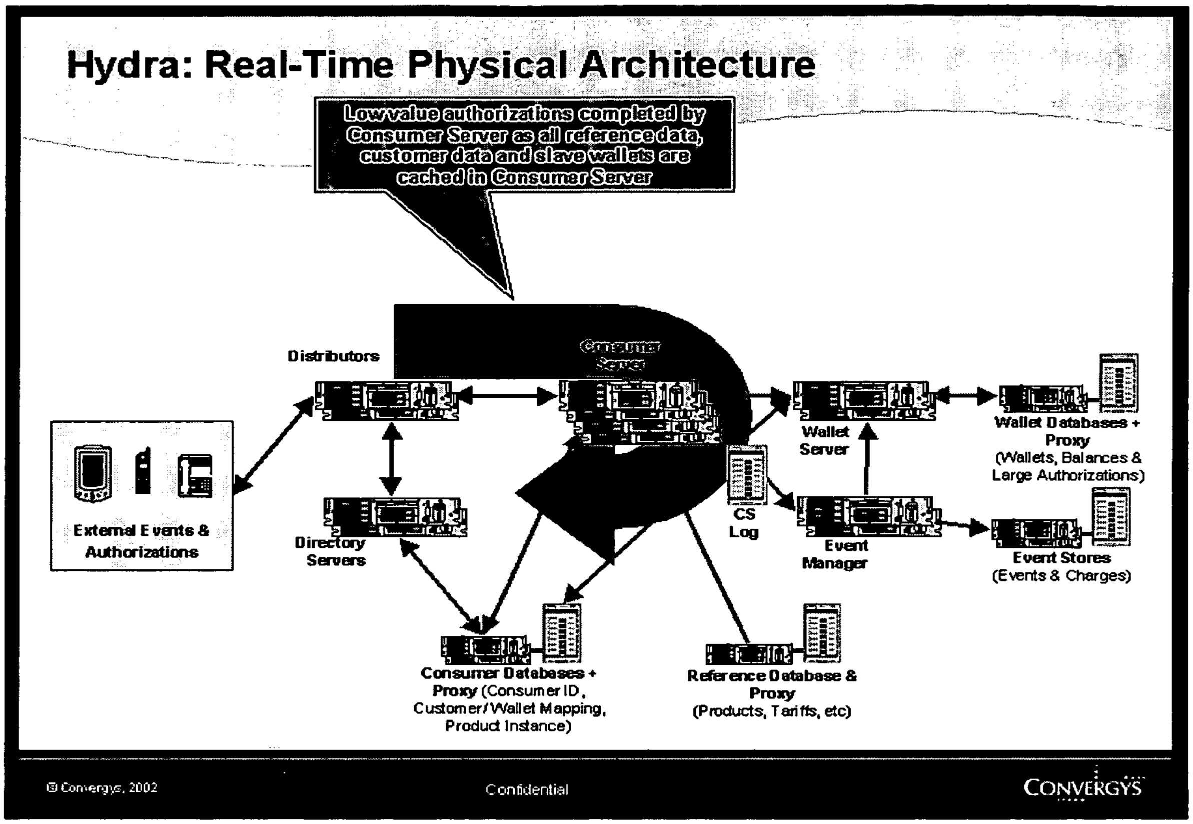
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 28



Hydra: Real-Time Physical Architecture

Low value authorizations completed by Consumer Server as all reference data, customer data and slave wallets are cached in Consumer Server

External Events & Authorizations

Distributors

Consumer Server

Directory Servers

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

CS Log

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

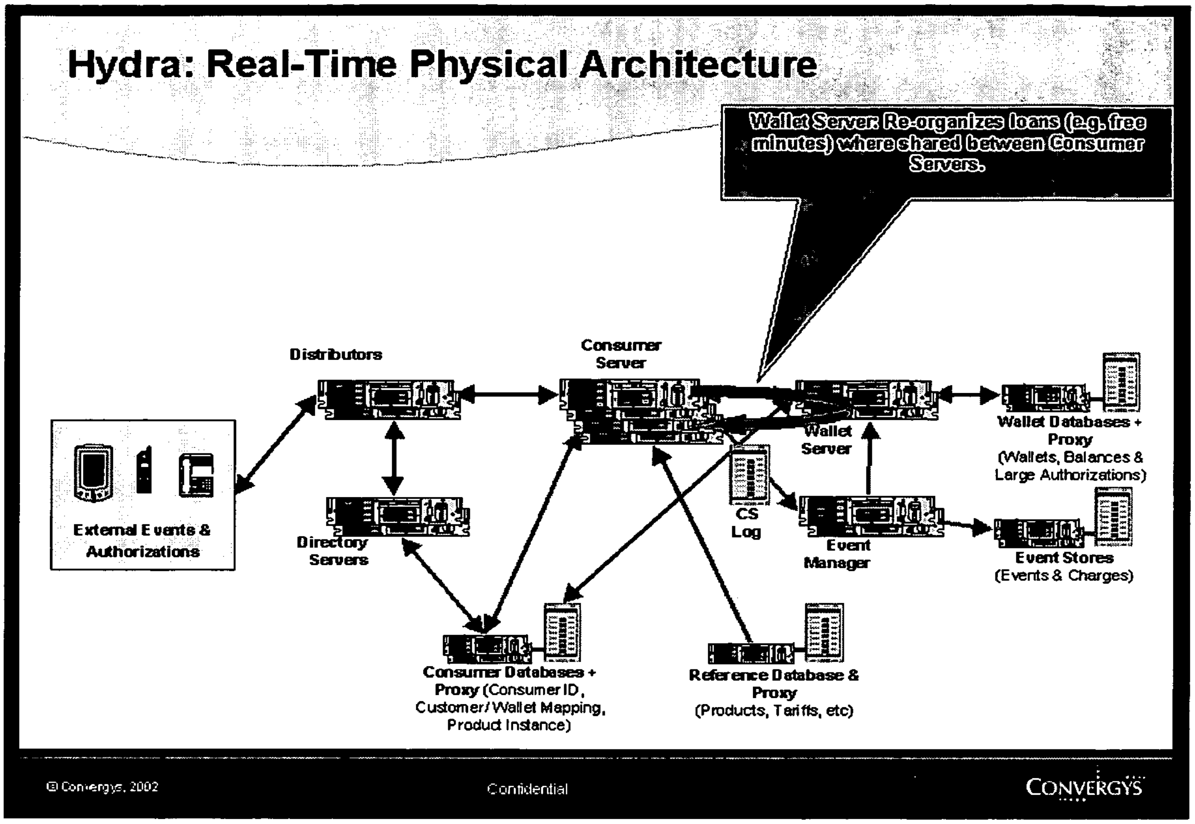
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 29



Hydra: Real-Time Physical Architecture

Wallet Server: Re-organizes loans (e.g. free minutes) where shared between Consumer Servers.

Distributors

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

External Events & Authorizations

Directory Servers

CS Log

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

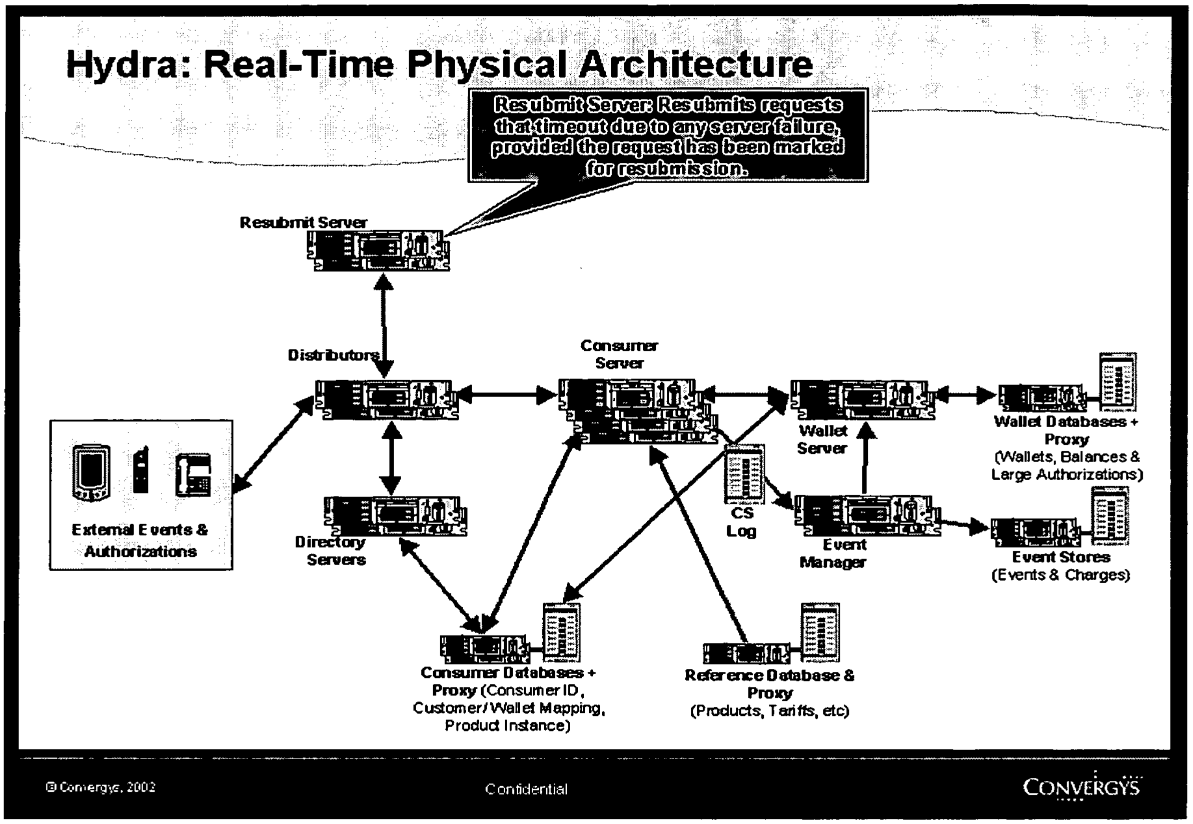
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 30



Hydra: Real-Time Physical Architecture

Resubmit Server: Resubmits requests that timeout due to any server failure, provided the request has been marked for resubmission.

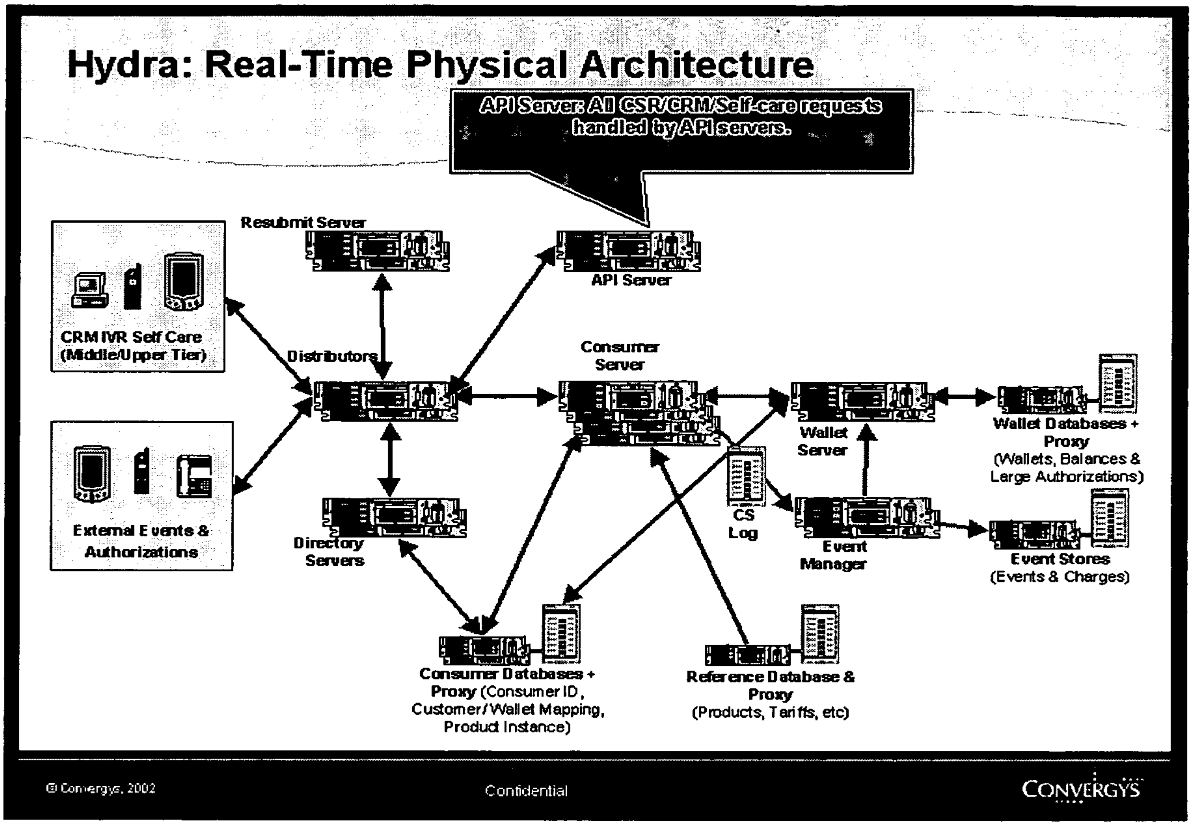
Resubmit Server
Distributors
Consumer Server
Wallet Server
Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)
External Events & Authorizations
Directory Servers
CS Log
Event Manager
Event Stores (Events & Charges)
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 31



Hydra: Real-Time Physical Architecture

API Server: All CSR/CRM/Self-care requests handled by API servers.

Resubmit Server

API Server

CRM IVR Self Care (Middle/Upper Tier)

Distributors

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

External Events & Authorizations

Directory Servers

CS Log

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

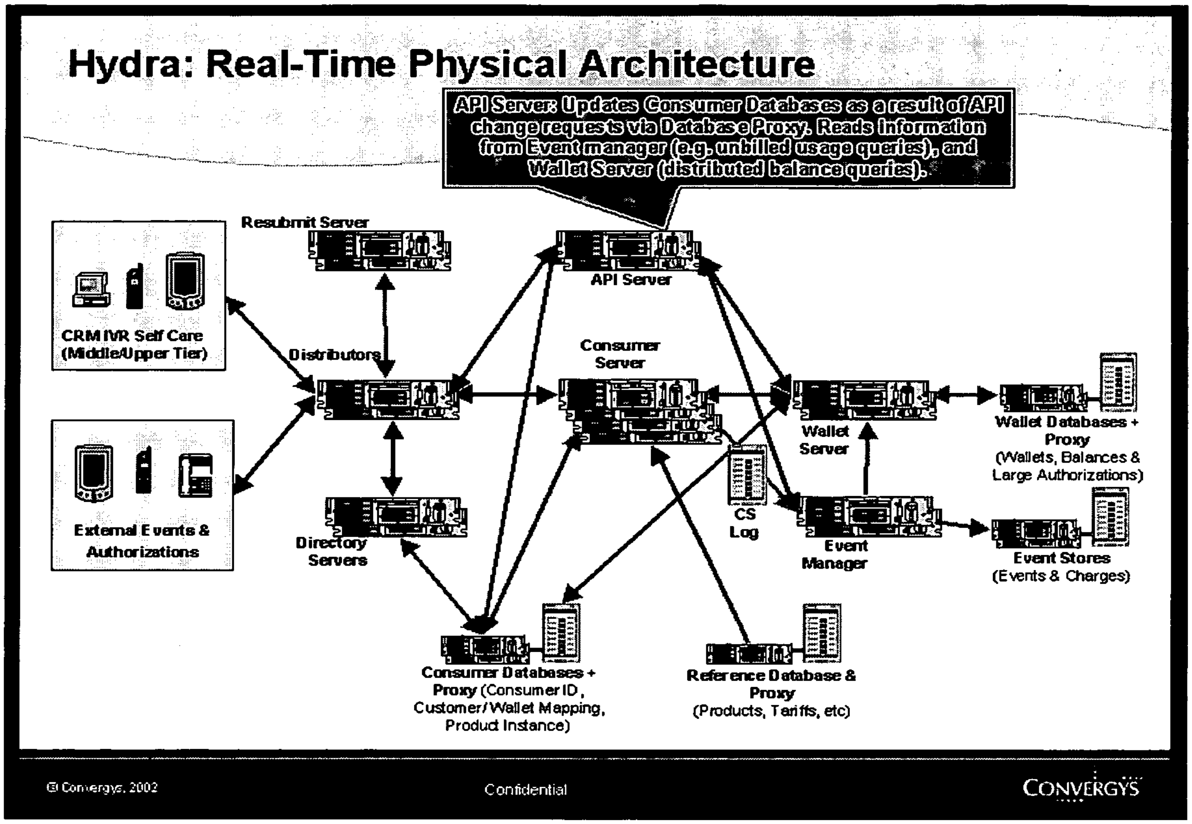
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 32



Hydra: Real-Time Physical Architecture

API Server: Updates Consumer Databases as a result of API change requests via Database Proxy. Reads information from Event manager (e.g. unbilled usage queries), and Wallet Server (distributed balance queries).

Resubmit Server

API Server

CRM IVR Self Care (Middle/Upper Tier)

Distributors

Consumer Server

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

External Events & Authorizations

Directory Servers

CS Log

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

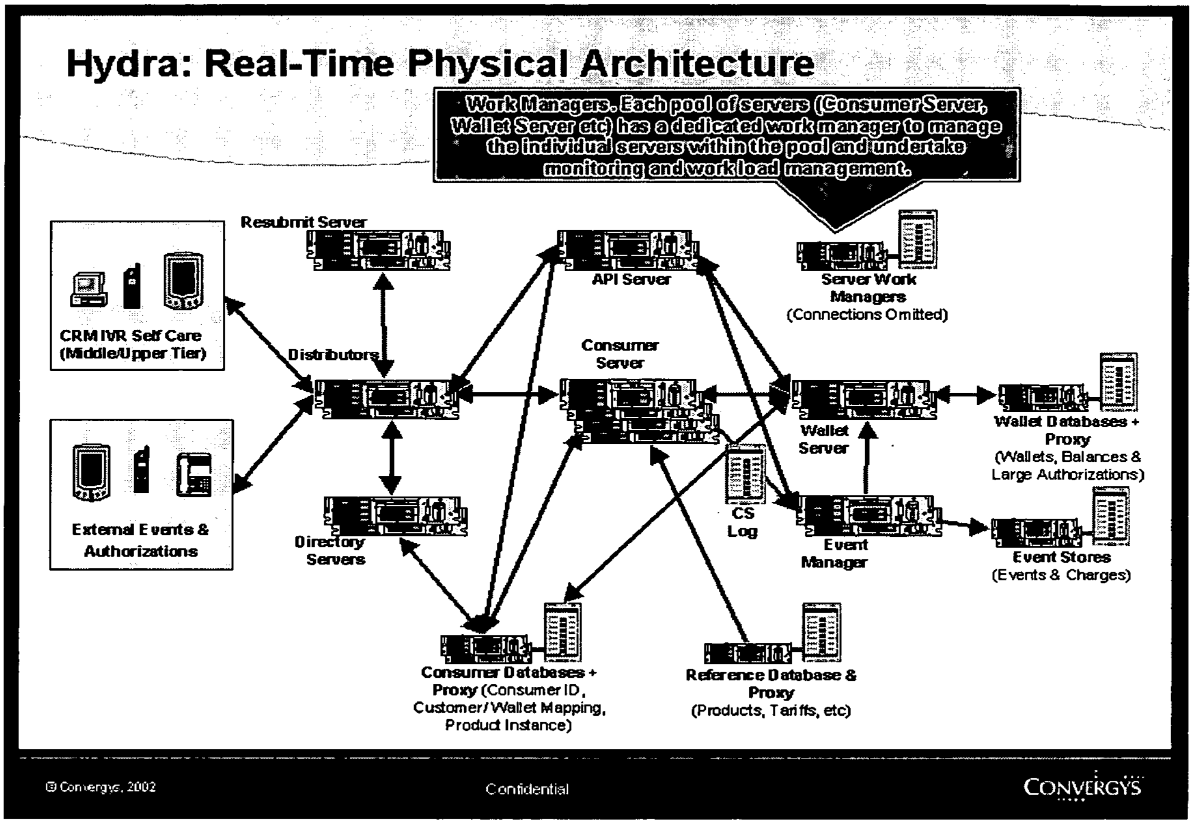
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 33



Hydra: Real-Time Physical Architecture

Work Managers. Each pool of servers (Consumer Server, Wallet Server etc) has a dedicated work manager to manage the individual servers within the pool and undertake monitoring and work load management.

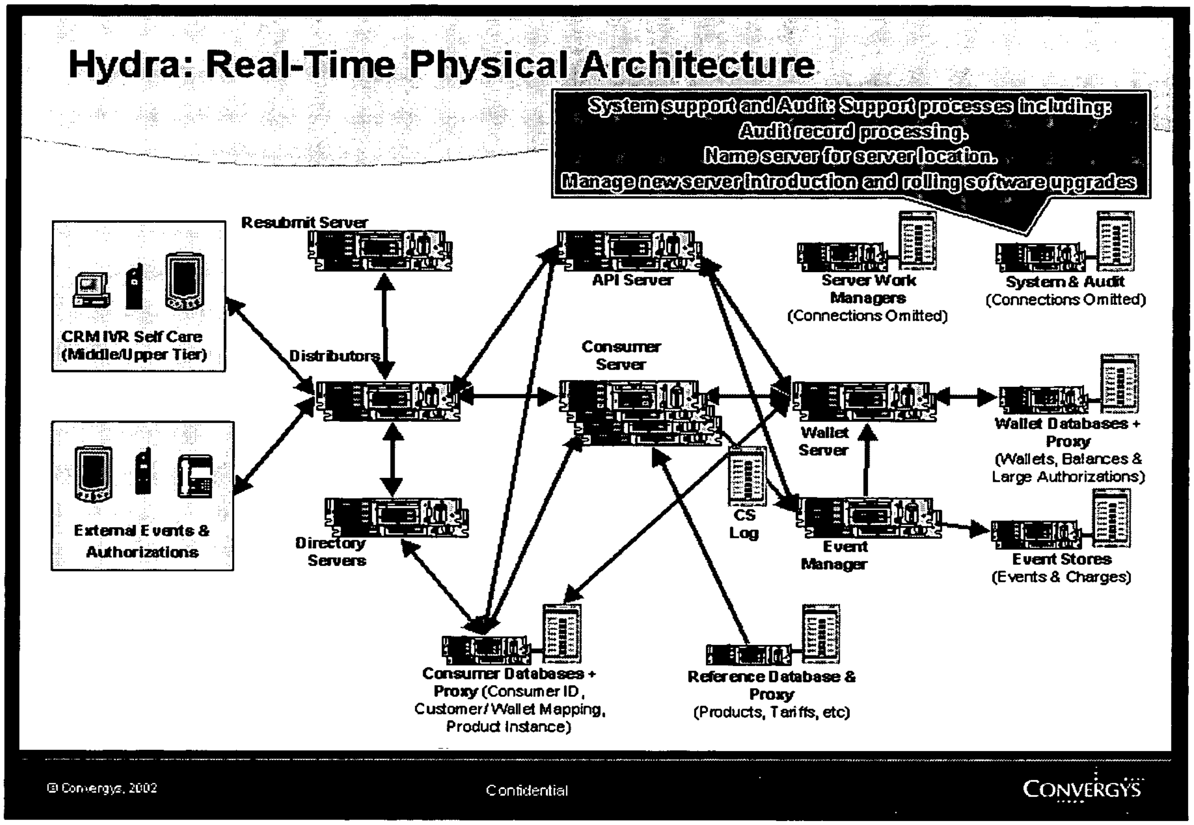
CRM IVR Self Care (Middle/Upper Tier)
External Events & Authorizations
Resubmit Server
Distributors
Directory Servers
API Server
Consumer Server
CS Log
Server Work Managers (Connections Omitted)
Wallet Server
Event Manager
Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)
Event Stores (Events & Charges)
Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 34



Hydra: Real-Time Physical Architecture

System support and Audit: Support processes including:
Audit record processing.
Name server for server location.
Manage new server introduction and rolling software upgrades

CRM IVR Self Care (Middle/Upper Tier)

External Events & Authorizations

Resubmit Server

Distributors

Directory Servers

API Server

Consumer Server

CS Log

Server Work Managers (Connections Omitted)

System & Audit (Connections Omitted)

Wallet Server

Wallet Databases + Proxy (Wallets, Balances & Large Authorizations)

Event Manager

Event Stores (Events & Charges)

Consumer Databases + Proxy (Consumer ID, Customer/Wallet Mapping, Product Instance)

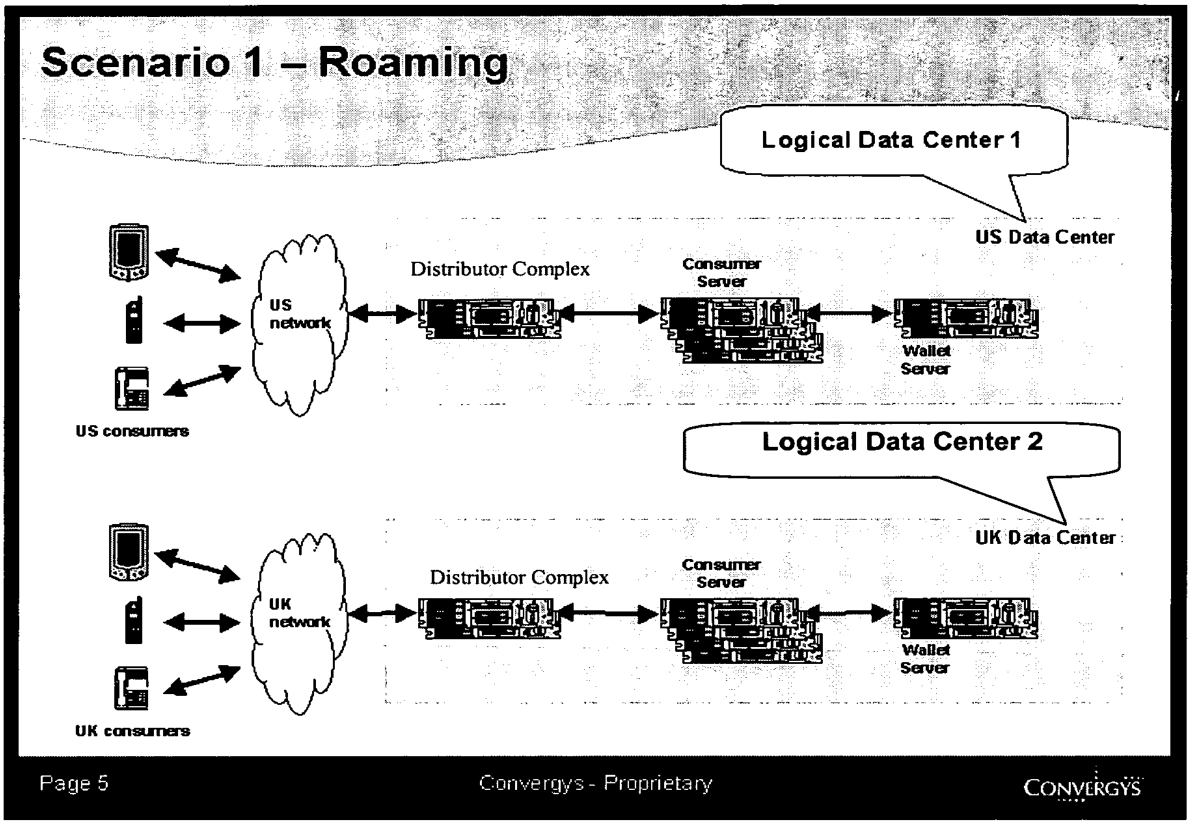
Reference Database & Proxy (Products, Tariffs, etc)

© Convergys, 2002

Confidential

CONVERGYS

Figure 35



Scenario 1 – Roaming

Logical Data Center 1

US Data Center

US network

Distributor Complex

Consumer Server

Wallet Server

US consumers

Logical Data Center 2

UK Data Center

UK network

Distributor Complex

Consumer Server

Wallet Server

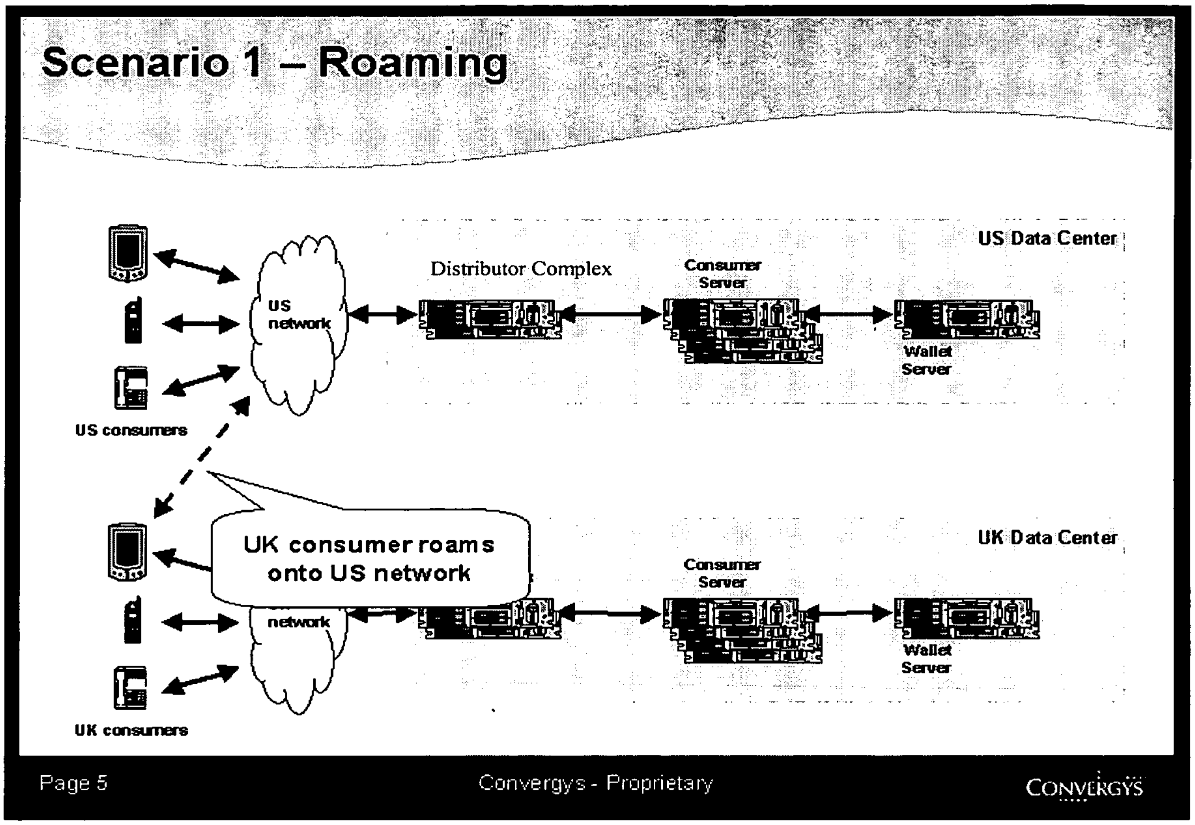
UK consumers

Page 5

Convergys - Proprietary

CONVERGYS

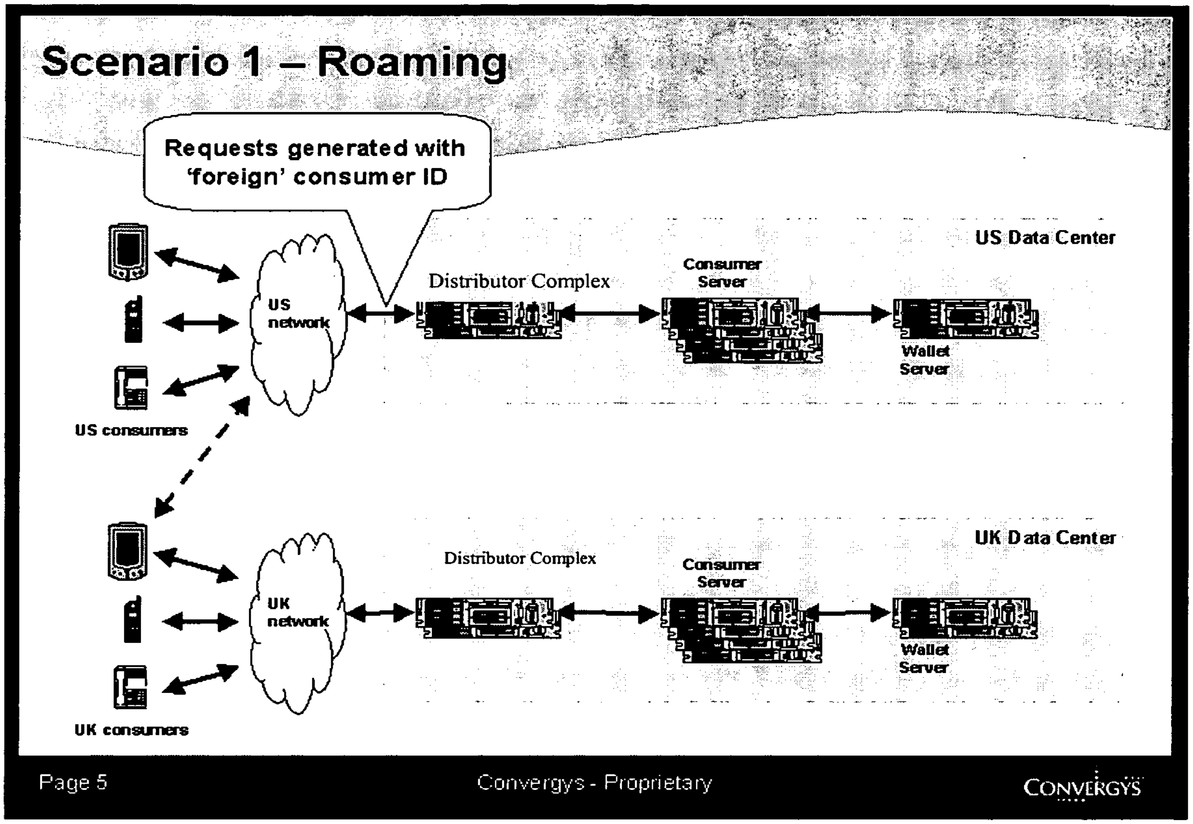
Figure 36



Scenario 1 – Roaming

Page 5 Convergys - Proprietary CONVERGYS

Figure 37



Scenario 1 – Roaming

Requests generated with 'foreign' consumer ID

US Data Center

US network

US consumers

Distributor Complex

Consumer Server

Wallet Server

UK Data Center

UK network

UK consumers

Distributor Complex

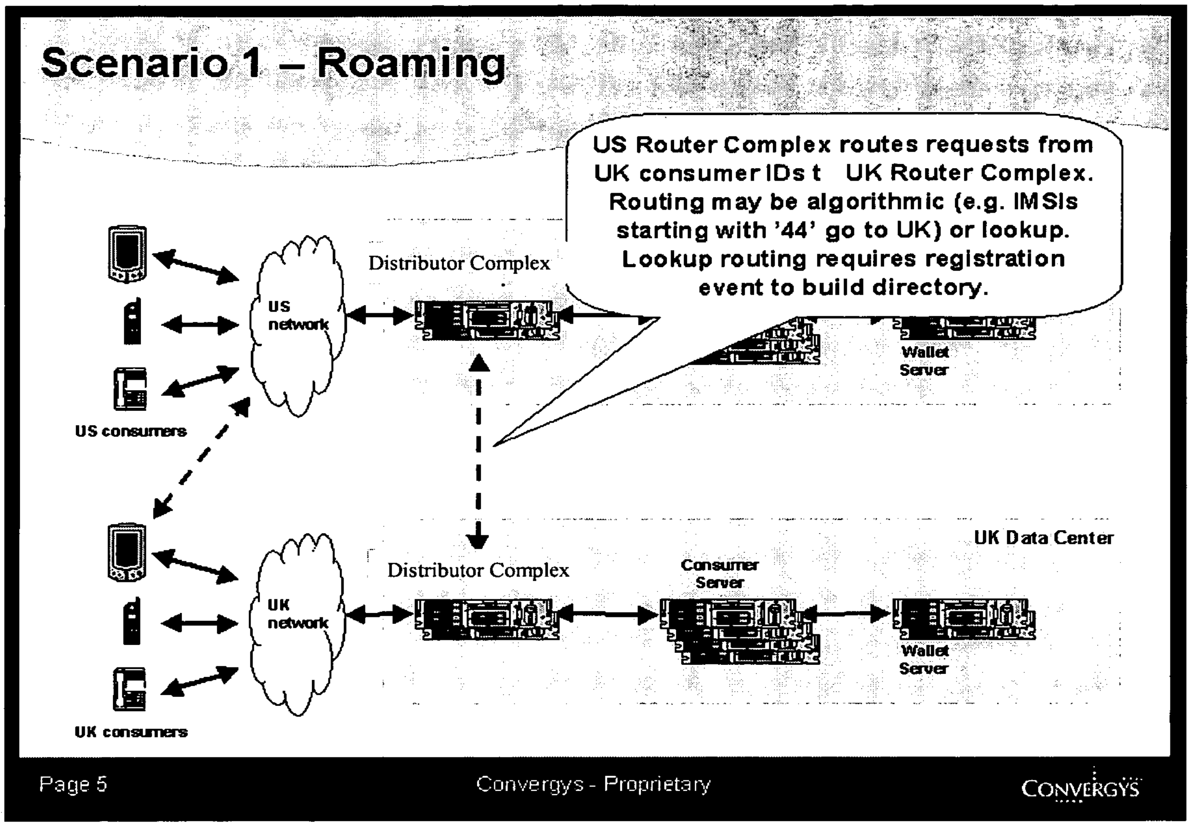
Consumer Server

Wallet Server

Convergys - Proprietary

CONVERGYS

Figure 38

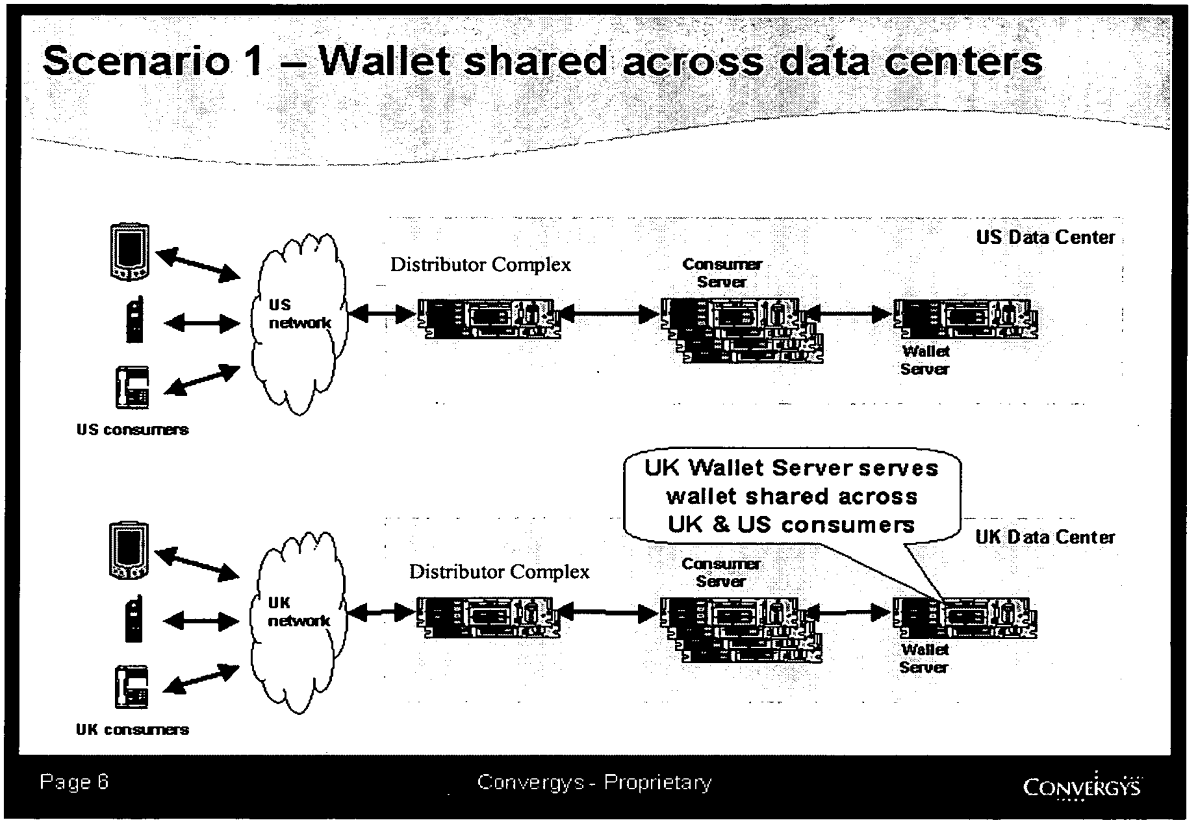


Scenario 1 – Roaming

US Router Complex routes requests from UK consumer IDs t UK Router Complex. Routing may be algorithmic (e.g. IMSIs starting with '44' go to UK) or lookup. Lookup routing requires registration event to build directory.

Page 5 Convergys - Proprietary CONVERGYS

Figure 39



Scenario 1 – Wallet shared across data centers

US Data Center

Distributor Complex

Consumer Server

Wallet Server

US network

US consumers

UK Wallet Server serves wallet shared across UK & US consumers

UK Data Center

Distributor Complex

Consumer Server

Wallet Server

UK network

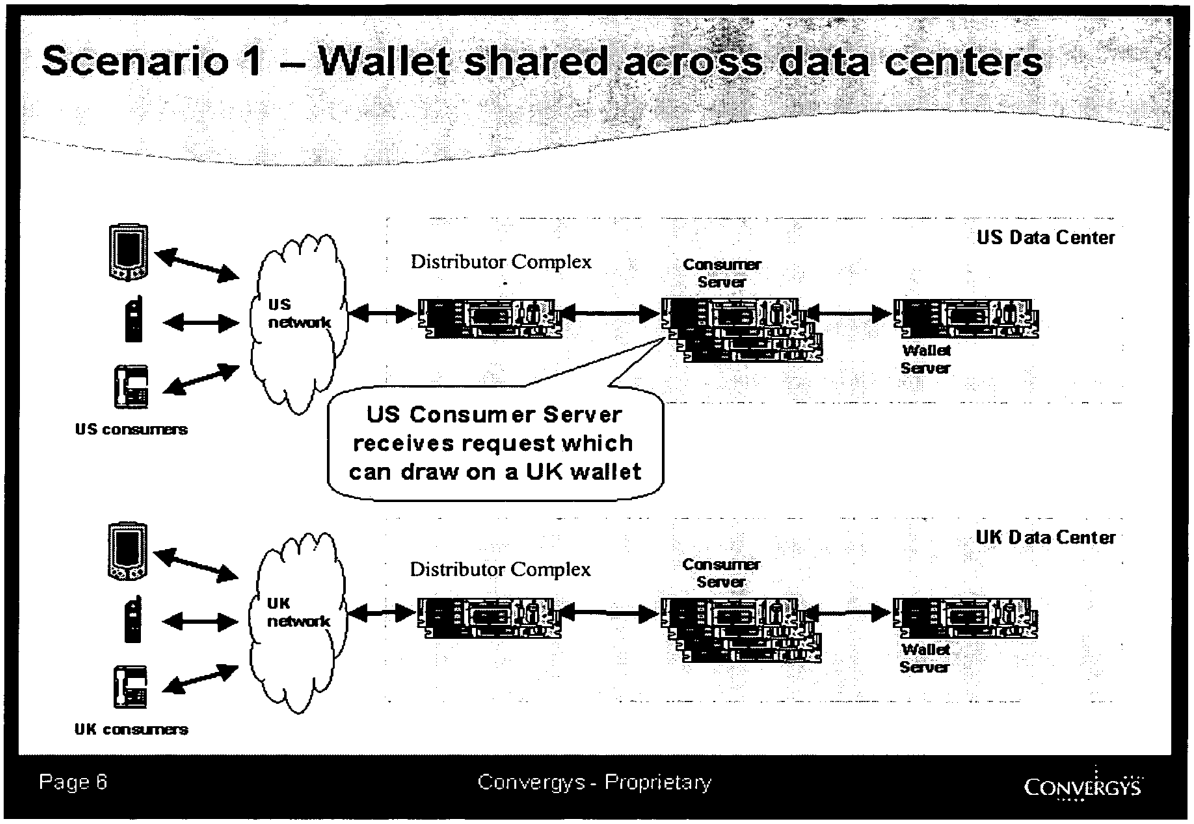
UK consumers

Page 6

Convergys - Proprietary

CONVERGYS

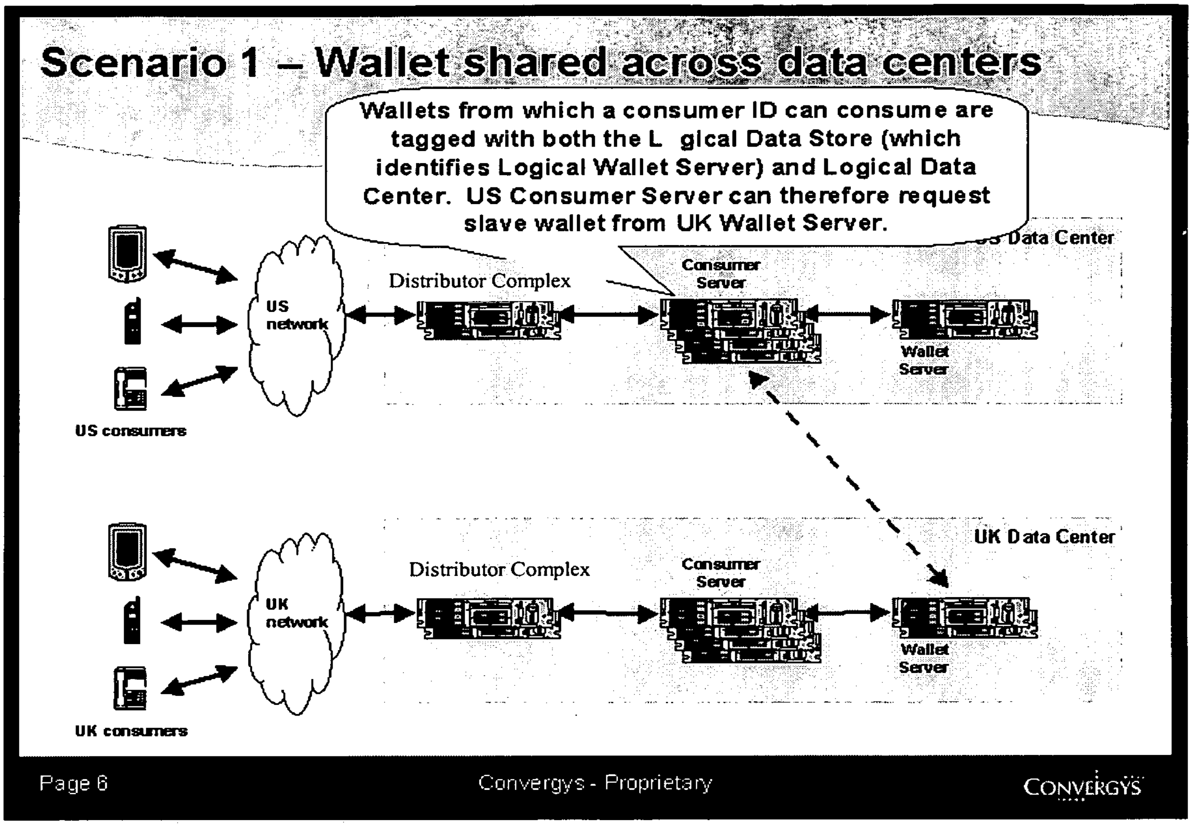
Figure 40



Scenario 1 – Wallet shared across data centers

US Consumer Server receives request which can draw on a UK wallet

Figure 41

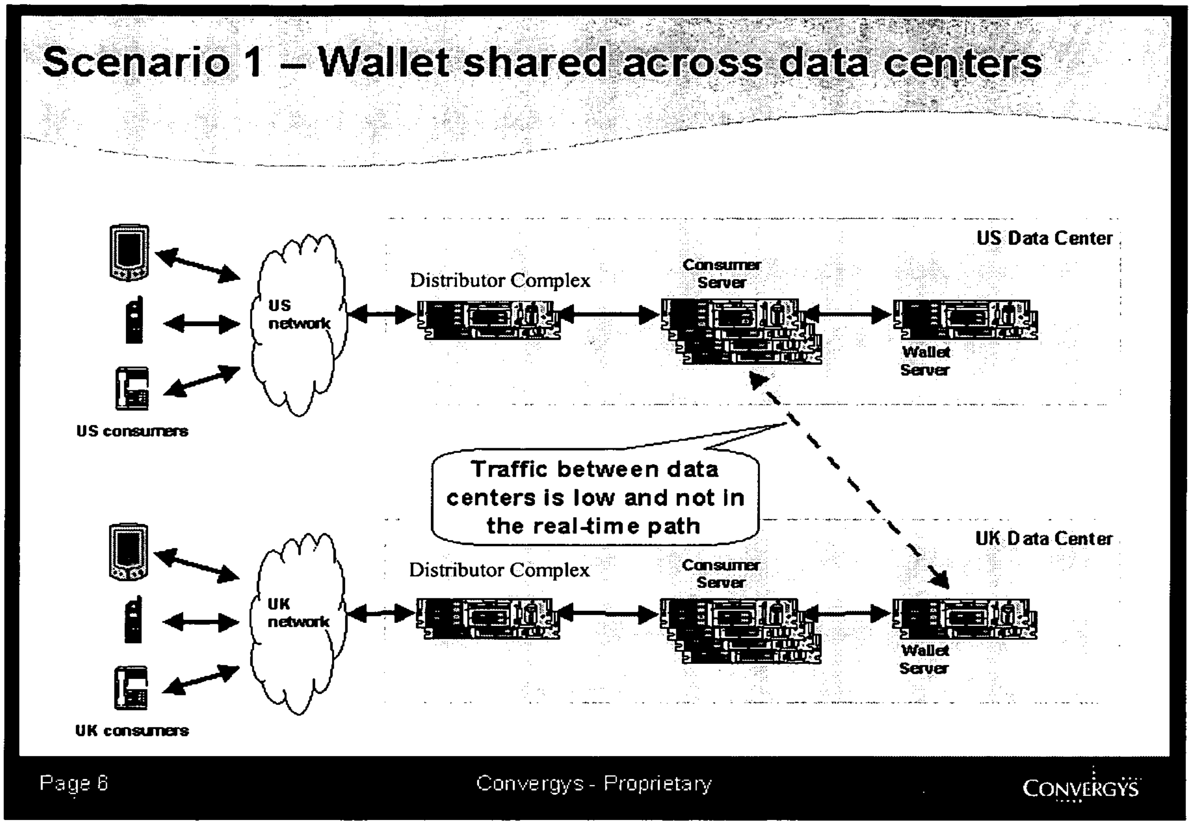


Scenario 1 – Wallet shared across data centers

Wallets from which a consumer ID can consume are tagged with both the Logical Data Store (which identifies Logical Wallet Server) and Logical Data Center. US Consumer Server can therefore request slave wallet from UK Wallet Server.

Page 6 Convergys - Proprietary CONVERGYS

Figure 42



Scenario 1 – Wallet shared across data centers

US Data Center

Distributor Complex

Consumer Server

Wallet Server

US network

US consumers

Traffic between data centers is low and not in the real-time path

UK Data Center

Distributor Complex

Consumer Server

Wallet Server

UK network

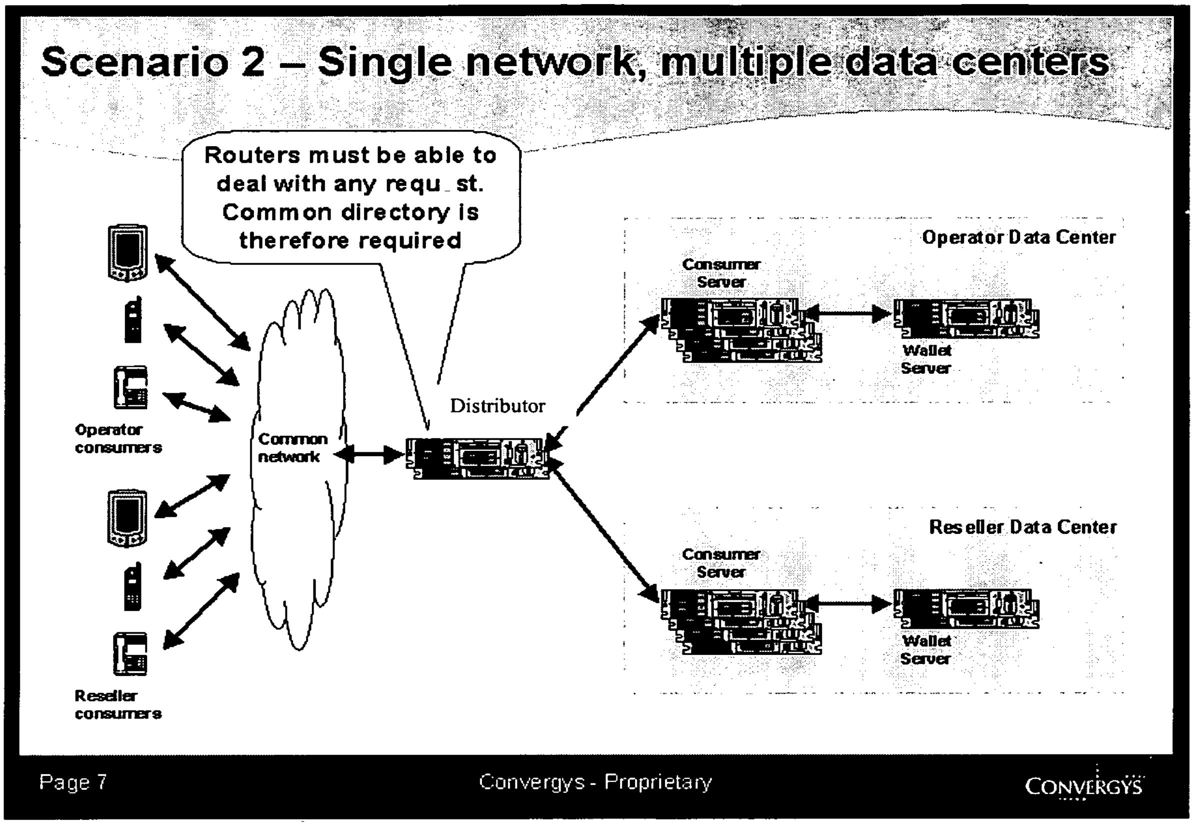
UK consumers

Page 6

Convergys - Proprietary

CONVERGYS

Figure 43



Scenario 2 – Single network, multiple data centers

Routers must be able to deal with any request. Common directory is therefore required

Page 7 Convergys - Proprietary CONVERGYS

Figure 44

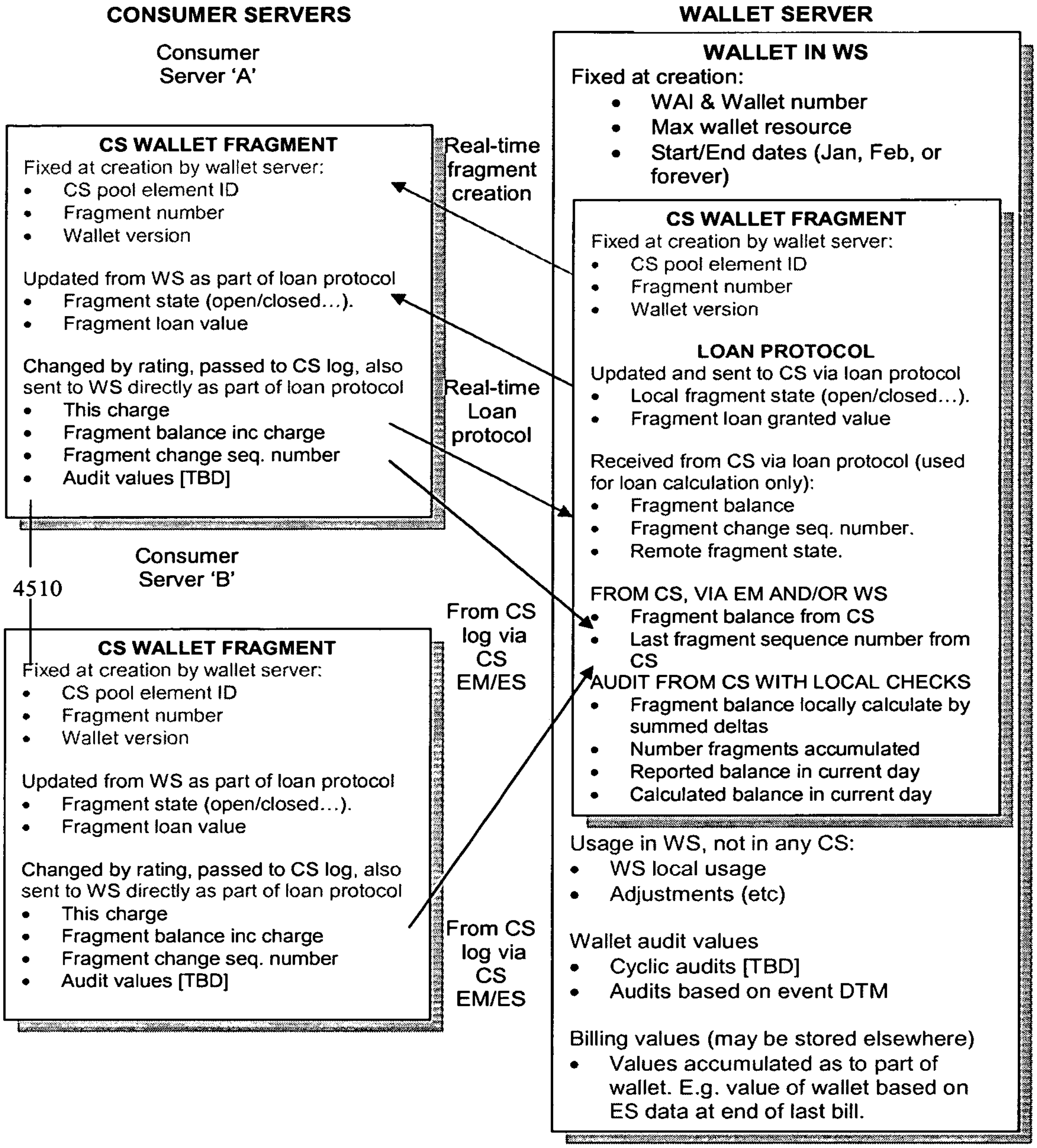

CONSUMER SERVERS
Consumer Server 'A'
CS WALLET FRAGMENT
Fixed at creation by wallet server:
• CS pool element ID
• Fragment number
• Wallet version
Updated from WS as part of loan protocol
• Fragment state (open/closed...).
• Fragment loan value
Changed by rating, passed to CS log, also sent to WS directly as part of loan protocol
• This charge
• Fragment balance inc charge
• Fragment change seq. number
• Audit values [TBD]
4510
Consumer Server 'B'
CS WALLET FRAGMENT
Fixed at creation by wallet server:
• CS pool element ID
• Fragment number
• Wallet version
Updated from WS as part of loan protocol
• Fragment state (open/closed...).
• Fragment loan value
Changed by rating, passed to CS log, also sent to WS directly as part of loan protocol
• This charge
• Fragment balance inc charge
• Fragment change seq. number
• Audit values [TBD]
Real-time fragment creation
Real-time Loan protocol
From CS log via CS EM/ES
From CS log via CS EM/ES
WALLET SERVER
WALLET IN WS
Fixed at creation:
• WAI & Wallet number
• Max wallet resource
• Start/End dates (Jan, Feb, or forever)
CS WALLET FRAGMENT
Fixed at creation by wallet server:
• CS pool element ID
• Fragment number
• Wallet version
LOAN PROTOCOL
Updated and sent to CS via loan protocol
• Local fragment state (open/closed...).
• Fragment loan granted value
Received from CS via loan protocol (used for loan calculation only):
• Fragment balance
• Fragment change seq. number.
• Remote fragment state.
FROM CS, VIA EM AND/OR WS
• Fragment balance from CS
• Last fragment sequence number from CS
AUDIT FROM CS WITH LOCAL CHECKS
• Fragment balance locally calculate by summed deltas
• Number fragments accumulated
• Reported balance in current day
• Calculated balance in current day
Usage in WS, not in any CS:
• WS local usage
• Adjustments (etc)
Wallet audit values
• Cyclic audits [TBD]
• Audits based on event DTM
Billing values (may be stored elsewhere)
• Values accumulated as to part of wallet. E.g. value of wallet based on ES data at end of last bill.

Figure 45A

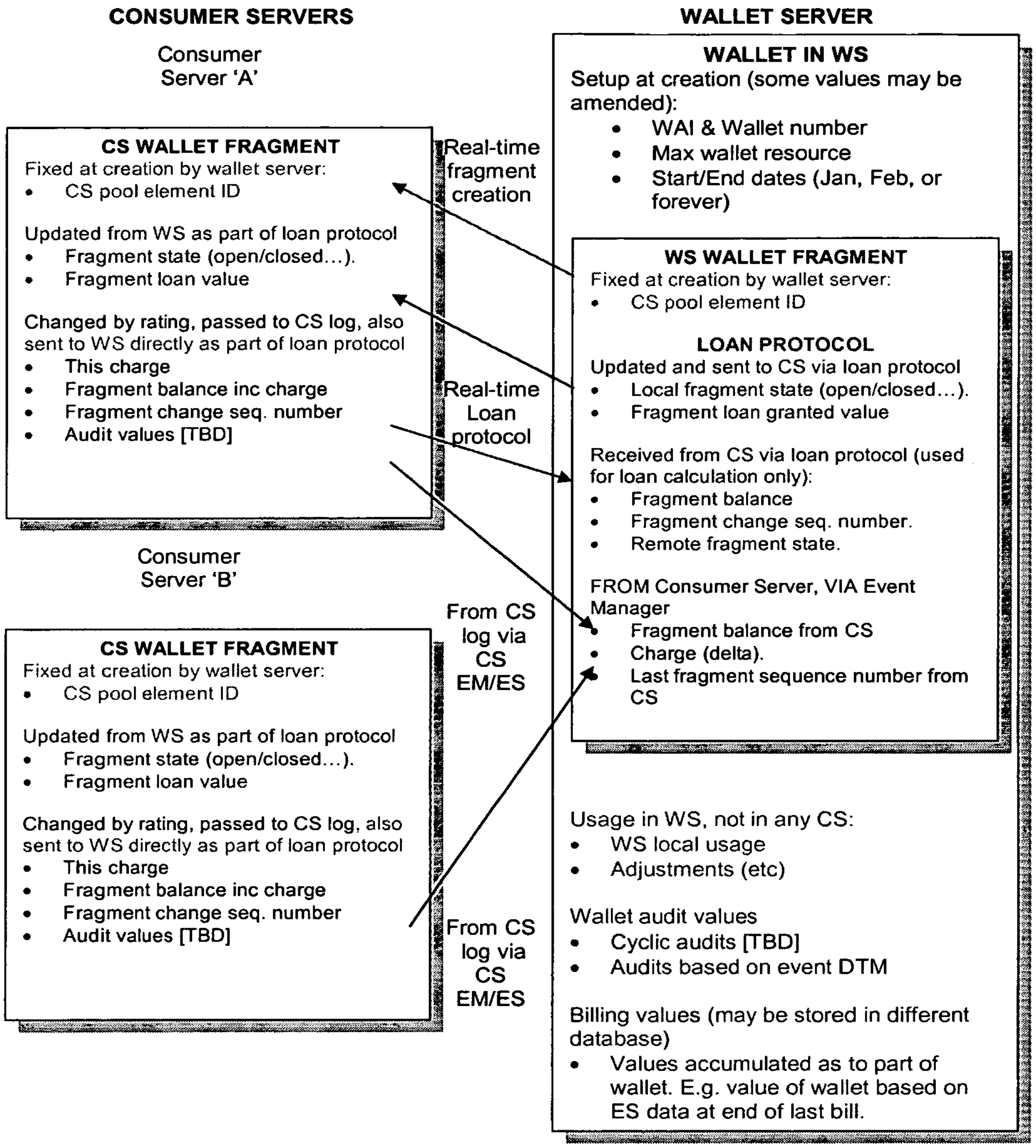

CONSUMER SERVERS
Consumer Server 'A'
CS WALLET FRAGMENT
Fixed at creation by wallet server:
CS pool element ID
Updated from WS as part of loan protocol
Fragment state (open/closed...).
Fragment loan value
Changed by rating, passed to CS log, also sent to WS directly as part of loan protocol
This charge
Fragment balance inc charge
Fragment change seq. number
Audit values [TBD]
Consumer Server 'B'
CS WALLET FRAGMENT
Fixed at creation by wallet server:
CS pool element ID
Updated from WS as part of loan protocol
Fragment state (open/closed...).
Fragment loan value
Changed by rating, passed to CS log, also sent to WS directly as part of loan protocol
This charge
Fragment balance inc charge
Fragment change seq. number
Audit values [TBD]
Real-time fragment creation
Real-time Loan protocol
From CS log via CS EM/ES
From CS log via CS EM/ES
WALLET SERVER
WALLET IN WS
Setup at creation (some values may be amended):
WAI & Wallet number
Max wallet resource
Start/End dates (Jan, Feb, or forever)
WS WALLET FRAGMENT
Fixed at creation by wallet server:
CS pool element ID
LOAN PROTOCOL
Updated and sent to CS via loan protocol
Local fragment state (open/closed...).
Fragment loan granted value
Received from CS via loan protocol (used for loan calculation only):
Fragment balance
Fragment change seq. number.
Remote fragment state.
FROM Consumer Server, VIA Event Manager
Fragment balance from CS
Charge (delta).
Last fragment sequence number from CS
Usage in WS, not in any CS:
WS local usage
Adjustments (etc)
Wallet audit values
Cyclic audits [TBD]
Audits based on event DTM
Billing values (may be stored in different database)
Values accumulated as to part of wallet. E.g. value of wallet based on ES data at end of last bill.

Figure 45B

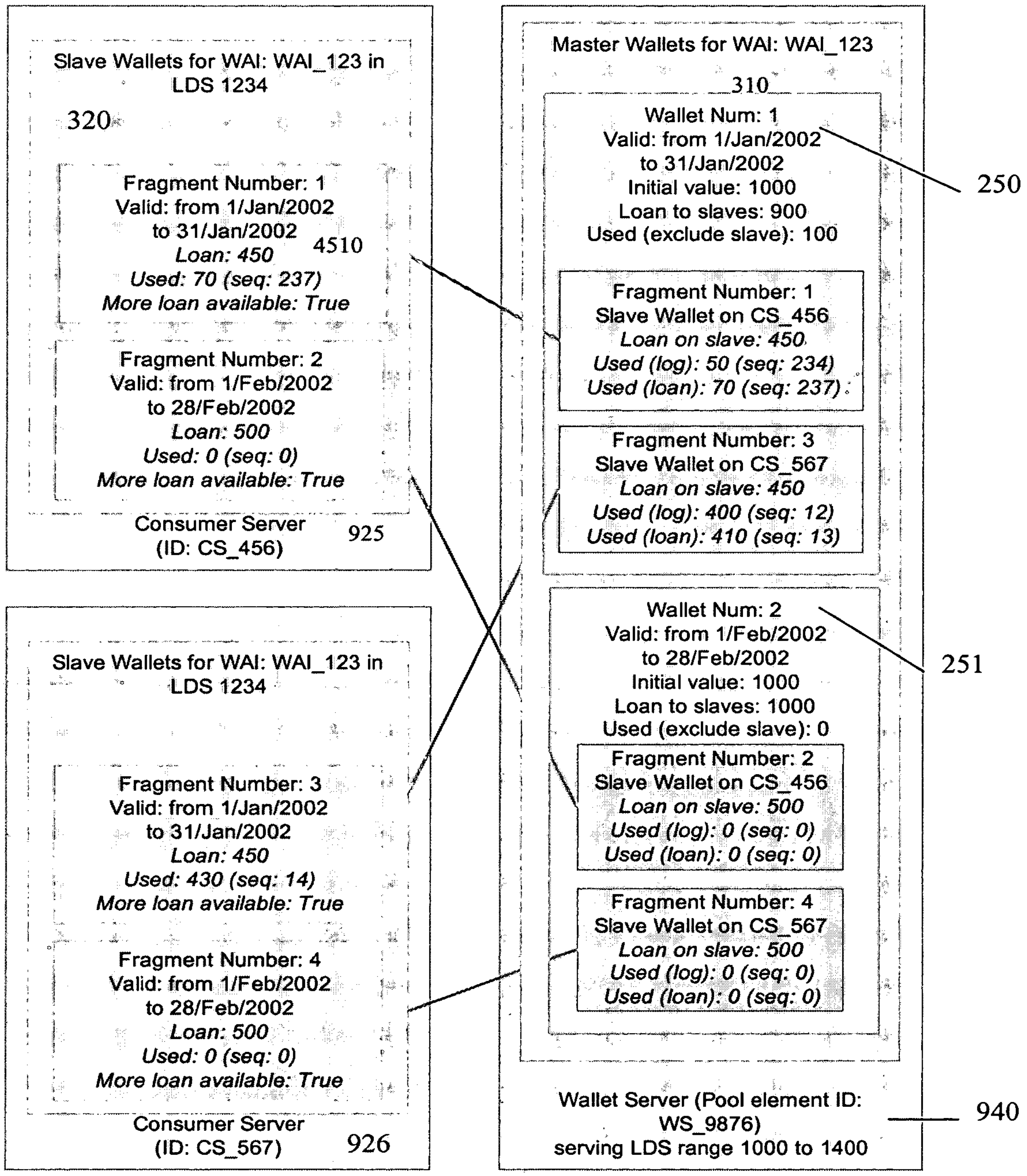

Slave Wallets for WAI: WAI_123 in LDS 1234
320
Fragment Number: 1
Valid: from 1/Jan/2002
to 31/Jan/2002
Loan: 450
4510
Used: 70 (seq: 237)
More loan available: True
Fragment Number: 2
Valid: from 1/Feb/2002
to 28/Feb/2002
Loan: 500
Used: 0 (seq: 0)
More loan available: True
Consumer Server
(ID: CS_456)
925
Slave Wallets for WAI: WAI_123 in LDS 1234
Fragment Number: 3
Valid: from 1/Jan/2002
to 31/Jan/2002
Loan: 450
Used: 430 (seq: 14)
More loan available: True
Fragment Number: 4
Valid: from 1/Feb/2002
to 28/Feb/2002
Loan: 500
Used: 0 (seq: 0)
More loan available: True
Consumer Server
(ID: CS_567)
926
Master Wallets for WAI: WAI_123
310
Wallet Num: 1
Valid: from 1/Jan/2002
to 31/Jan/2002
Initial value: 1000
Loan to slaves: 900
Used (exclude slave): 100
250
Fragment Number: 1
Slave Wallet on CS_456
Loan on slave: 450
Used (log): 50 (seq: 234)
Used (loan): 70 (seq: 237)
Fragment Number: 3
Slave Wallet on CS_567
Loan on slave: 450
Used (log): 400 (seq: 12)
Used (loan): 410 (seq: 13)
Wallet Num: 2
Valid: from 1/Feb/2002
to 28/Feb/2002
Initial value: 1000
Loan to slaves: 1000
Used (exclude slave): 0
251
Fragment Number: 2
Slave Wallet on CS_456
Loan on slave: 500
Used (log): 0 (seq: 0)
Used (loan): 0 (seq: 0)
Fragment Number: 4
Slave Wallet on CS_567
Loan on slave: 500
Used (log): 0 (seq: 0)
Used (loan): 0 (seq: 0)
Wallet Server (Pool element ID: WS_9876)
serving LDS range 1000 to 1400
940

Figure 46A

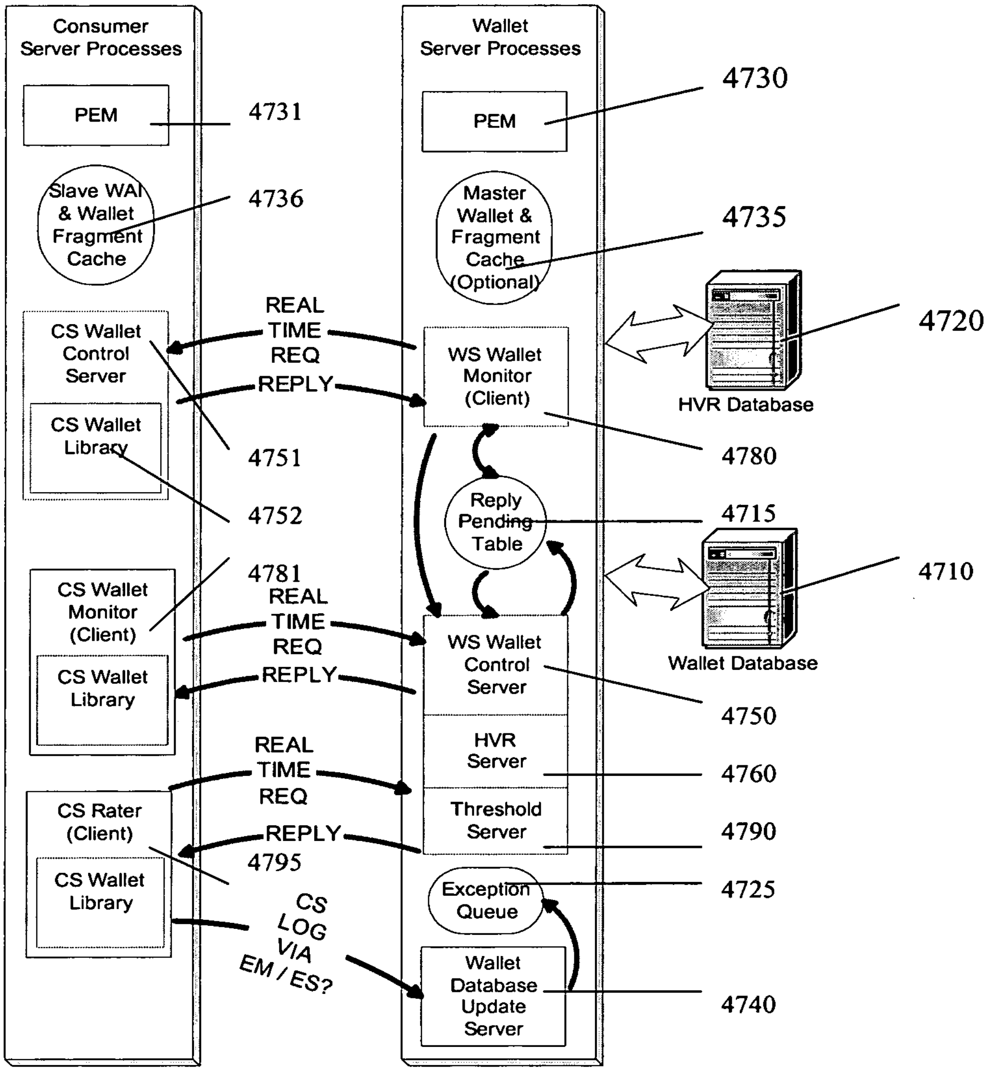

Consumer Server Processes
PEM
4731
Slave WAI & Wallet Fragment Cache
4736
CS Wallet Control Server
CS Wallet Library
4751
4752
CS Wallet Monitor (Client)
CS Wallet Library
4781
CS Rater (Client)
CS Wallet Library
4795
REAL TIME REQ
REPLY
REAL TIME REQ
REPLY
REAL TIME REQ
REPLY
CS LOG VIA EM / ES?
Wallet Server Processes
PEM
4730
Master Wallet & Fragment Cache (Optional)
4735
WS Wallet Monitor (Client)
4780
Reply Pending Table
4715
WS Wallet Control Server
4750
HVR Server
4760
Threshold Server
4790
Exception Queue
4725
Wallet Database Update Server
4740
HVR Database
4720
Wallet Database
4710

Figure 47A

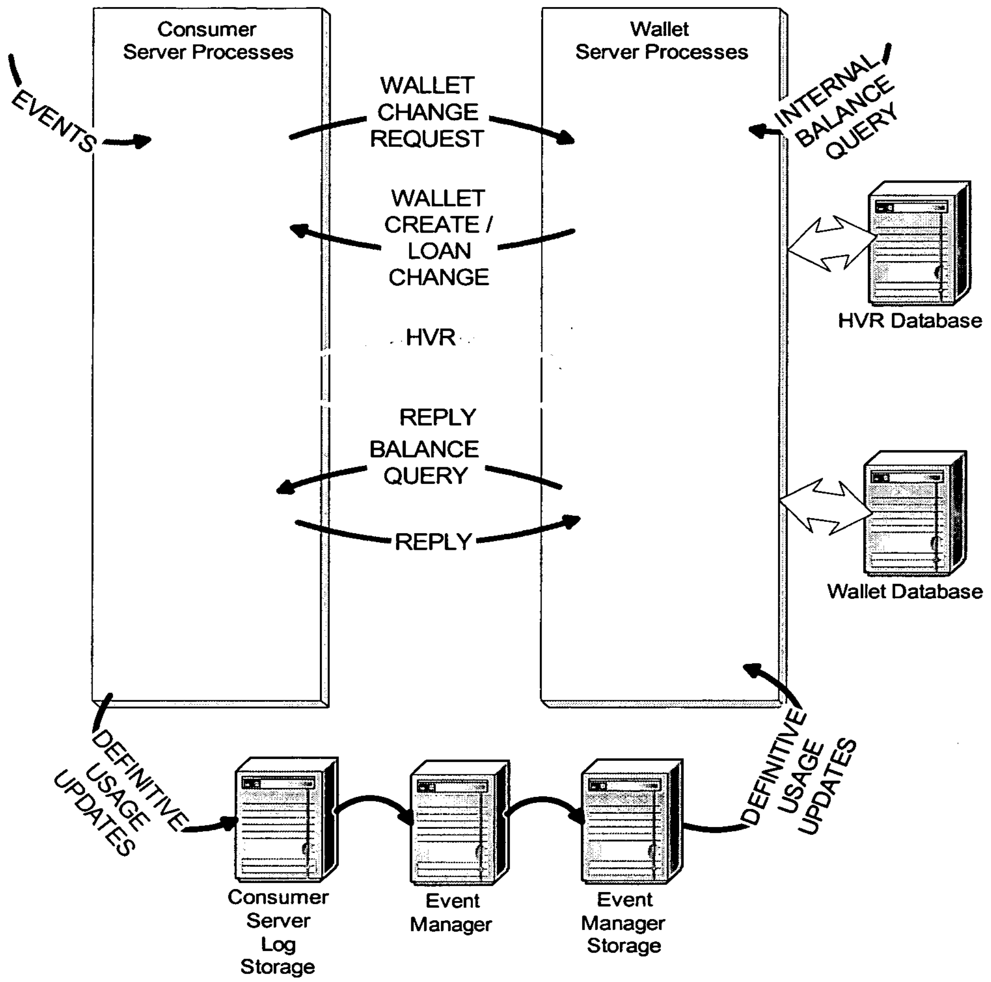

Consumer Server Processes
Wallet Server Processes
EVENTS
WALLET CHANGE REQUEST
INTERNAL BALANCE QUERY
WALLET CREATE / LOAN CHANGE
HVR
HVR Database
REPLY
BALANCE QUERY
REPLY
Wallet Database
DEFINITIVE USAGE UPDATES
DEFINITIVE USAGE UPDATES
Consumer Server Log Storage
Event Manager
Event Manager Storage

Figure 47B

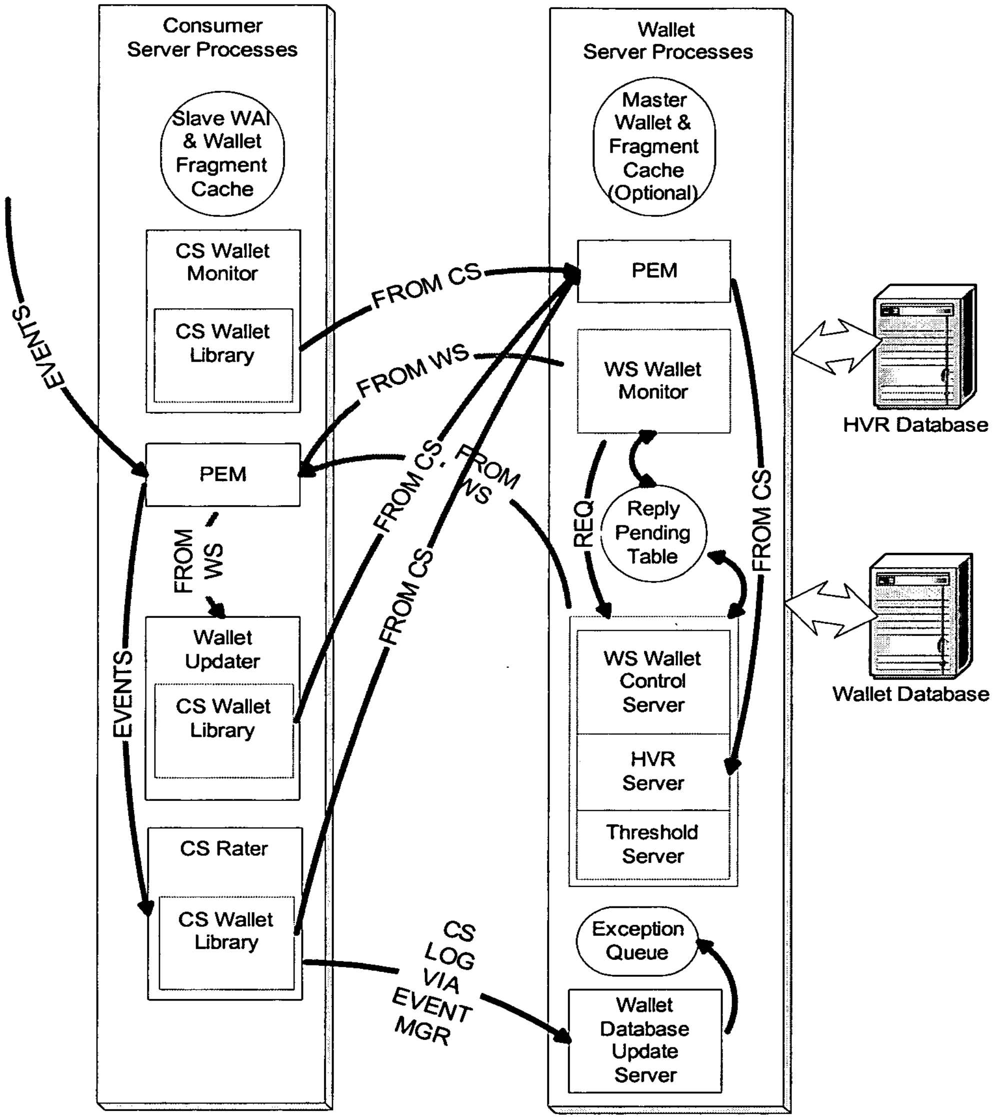

Consumer Server Processes
Slave WAI & Wallet Fragment Cache
CS Wallet Monitor
CS Wallet Library
PEM
Wallet Updater
CS Wallet Library
CS Rater
CS Wallet Library
EVENTS
EVENTS
FROM WS
FROM CS
FROM WS
FROM CS
FROM WS
FROM CS
CS LOG VIA EVENT MGR
Wallet Server Processes
Master Wallet & Fragment Cache (Optional)
PEM
WS Wallet Monitor
REQ
Reply Pending Table
FROM CS
WS Wallet Control Server
HVR Server
Threshold Server
Exception Queue
Wallet Database Update Server
HVR Database
Wallet Database

Figure 47C

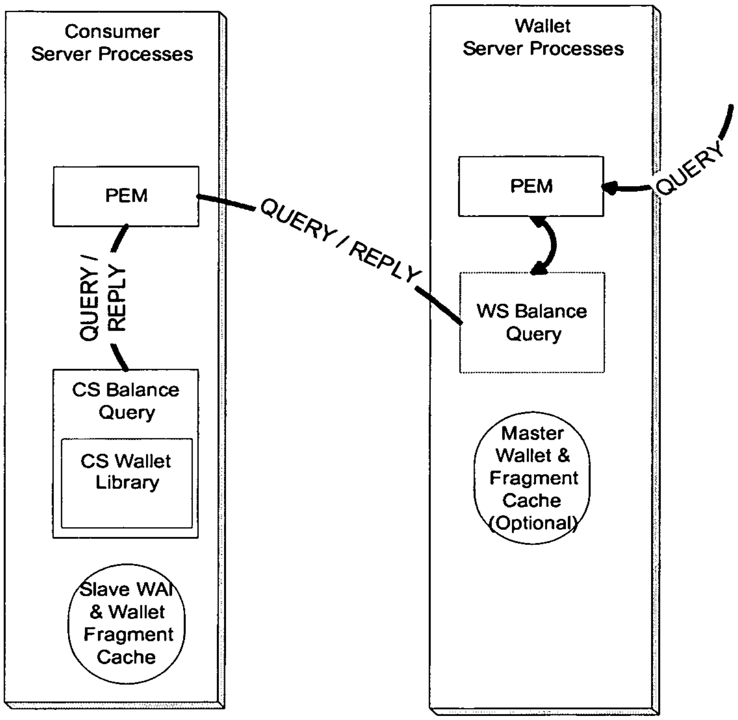

Consumer Server Processes
PEM
QUERY / REPLY
CS Balance Query
CS Wallet Library
Slave WAI & Wallet Fragment Cache
QUERY / REPLY
Wallet Server Processes
PEM
QUERY
WS Balance Query
Master Wallet & Fragment Cache (Optional)

Figure 47D

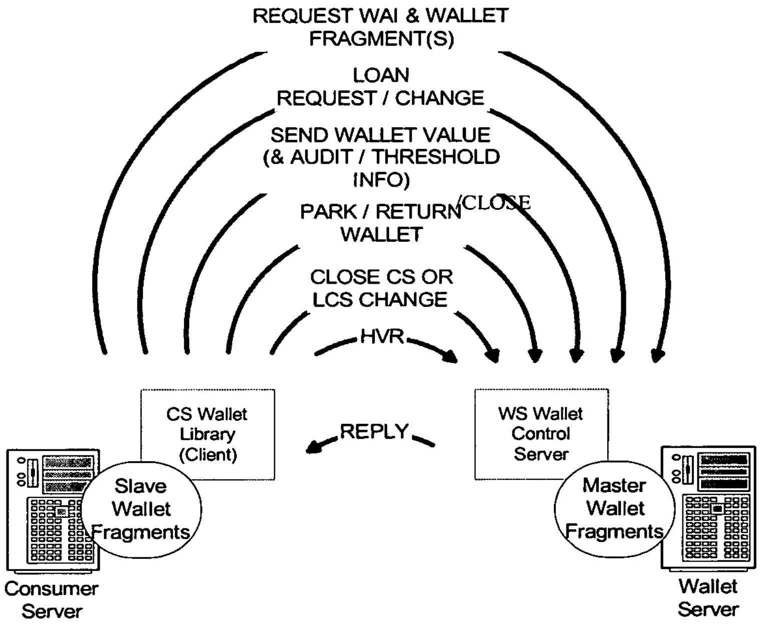

REQUEST WAI & WALLET FRAGMENT(S)
LOAN REQUEST / CHANGE
SEND WALLET VALUE (& AUDIT / THRESHOLD INFO)
PARK / RETURN/CLOSE WALLET
CLOSE CS OR LCS CHANGE
HVR
REPLY
CS Wallet Library (Client)
WS Wallet Control Server
Slave Wallet Fragments
Master Wallet Fragments
Consumer Server
Wallet Server

Figure 48A

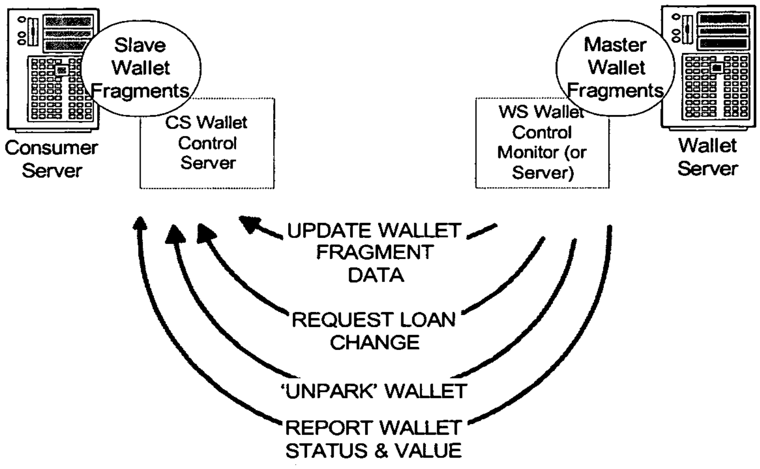


Figure 48B

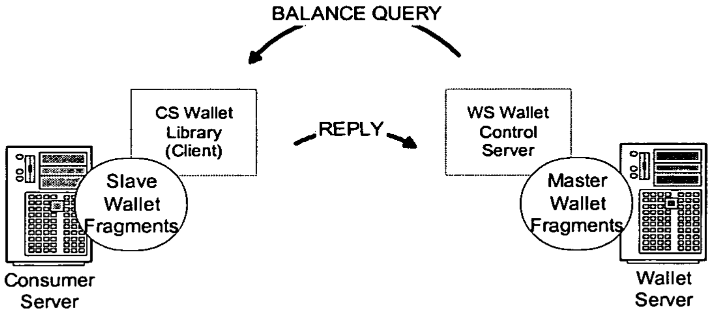


Figure 48C

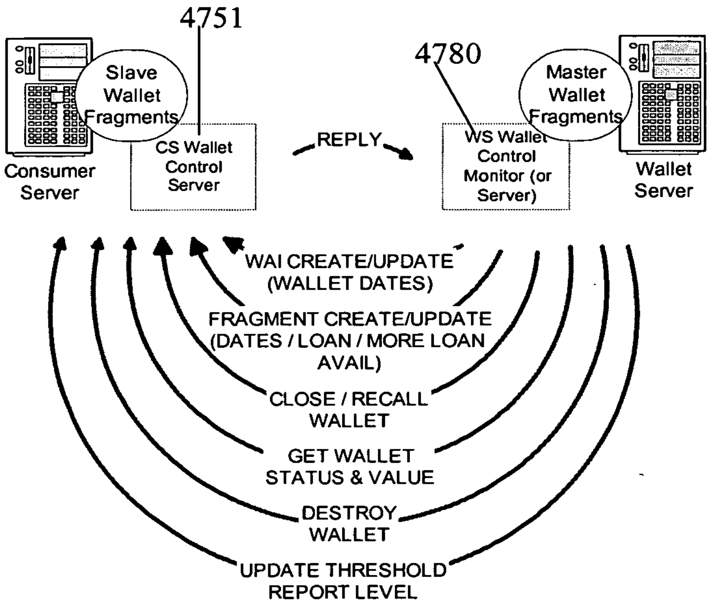

4751
4780
Slave Wallet Fragments
Consumer Server
CS Wallet Control Server
REPLY
WS Wallet Control Monitor (or Server)
Master Wallet Fragments
Wallet Server
WAI CREATE/UPDATE (WALLET DATES)
FRAGMENT CREATE/UPDATE (DATES / LOAN / MORE LOAN AVAIL)
CLOSE / RECALL WALLET
GET WALLET STATUS & VALUE
DESTROY WALLET
UPDATE THRESHOLD REPORT LEVEL

Figure 49

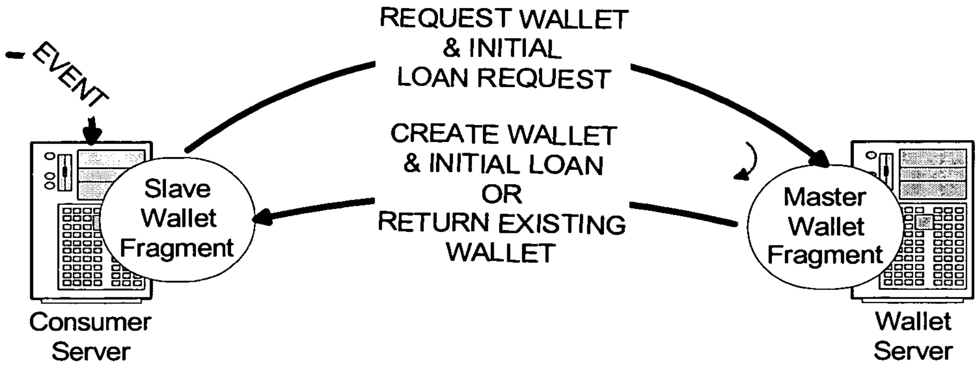


Figure 50A

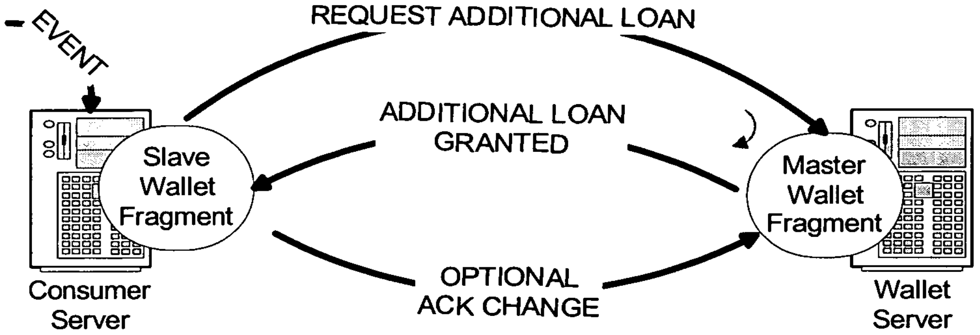


Figure 50B

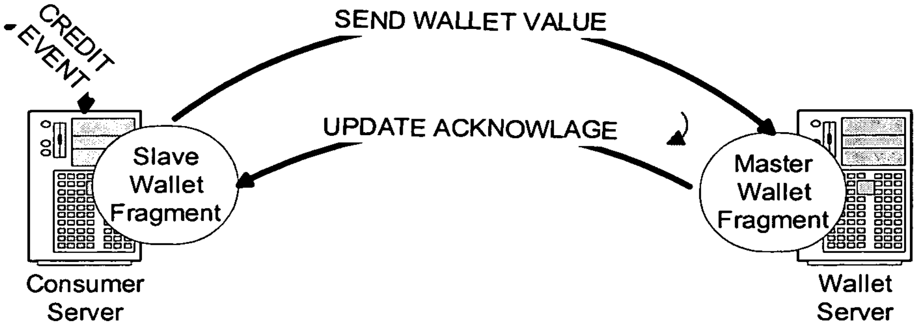


Figure 50C

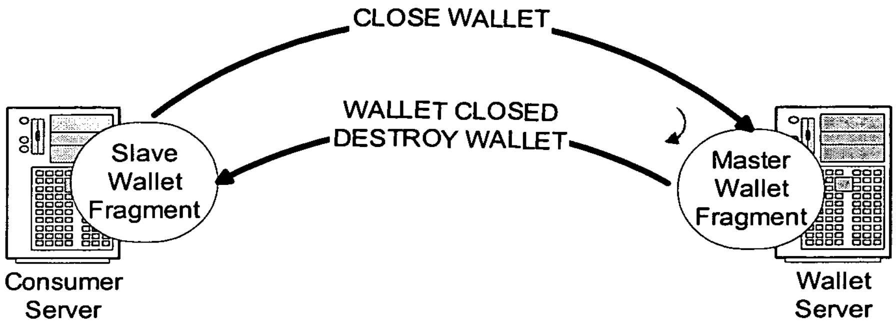


Figure 50D

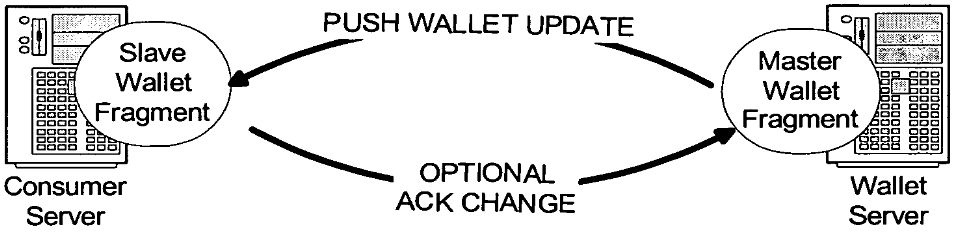


Figure 50E

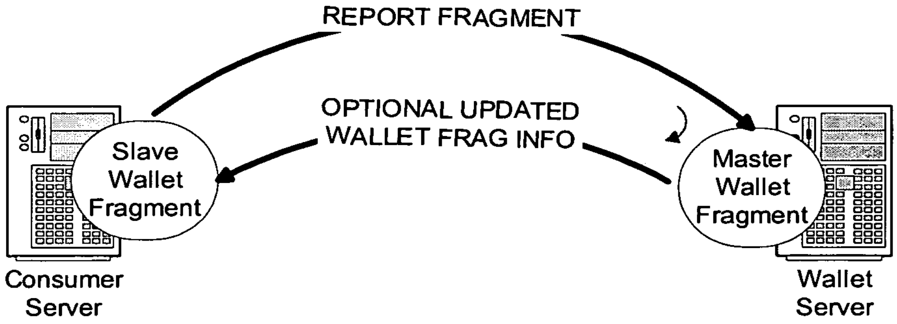


Figure 50F

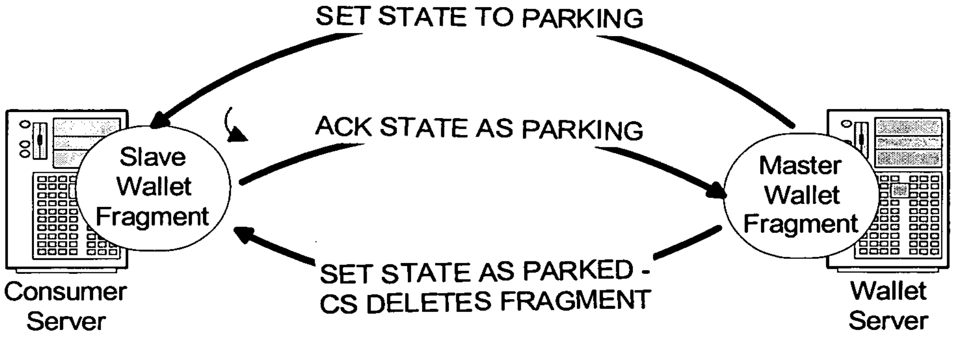


Figure 50G

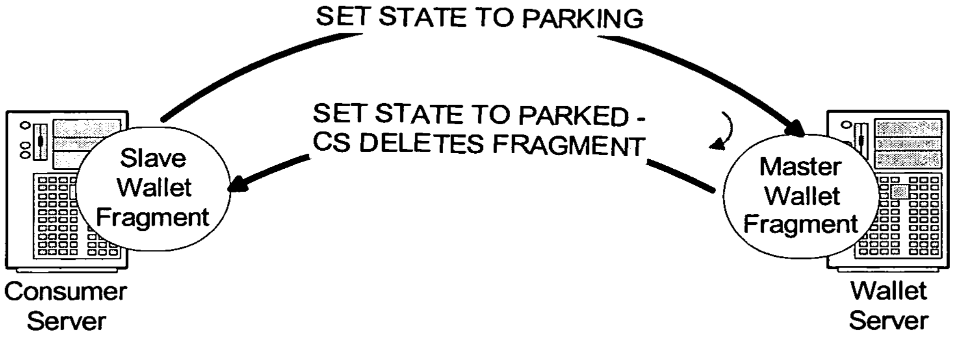


Figure 50H

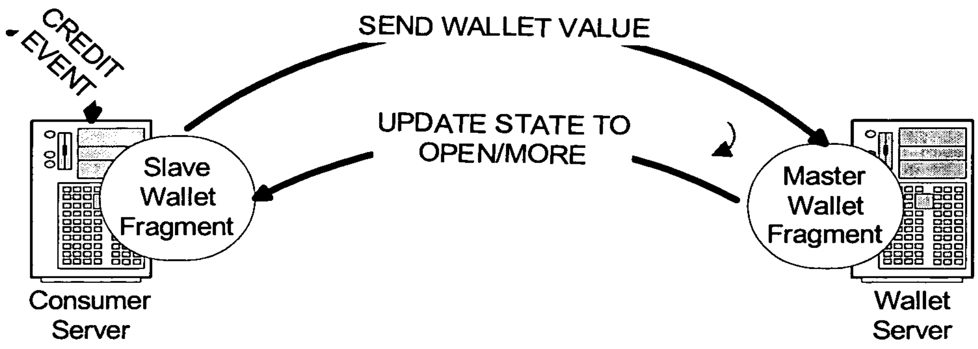


Figure 50I

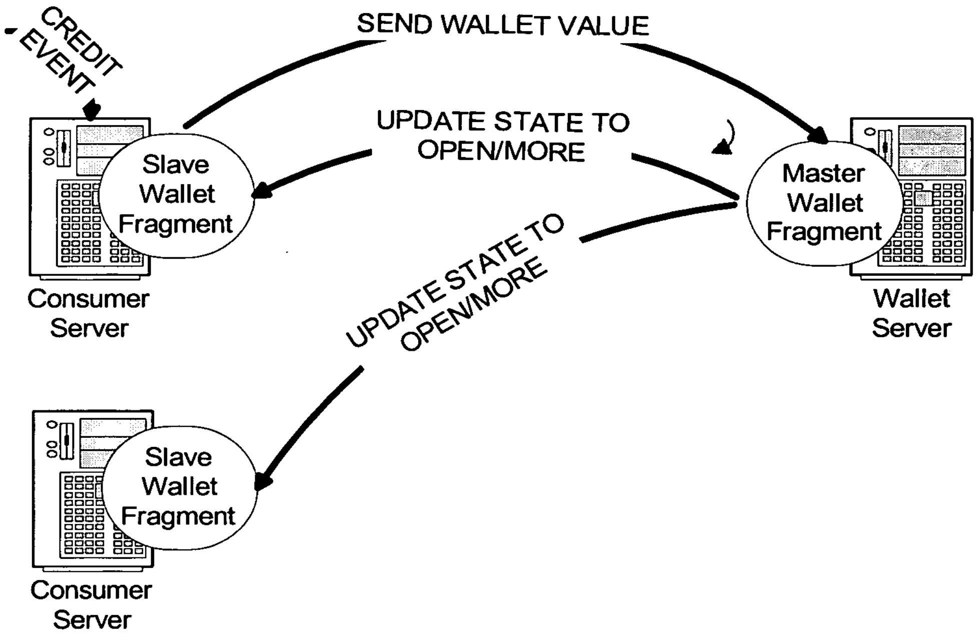


Figure 50J

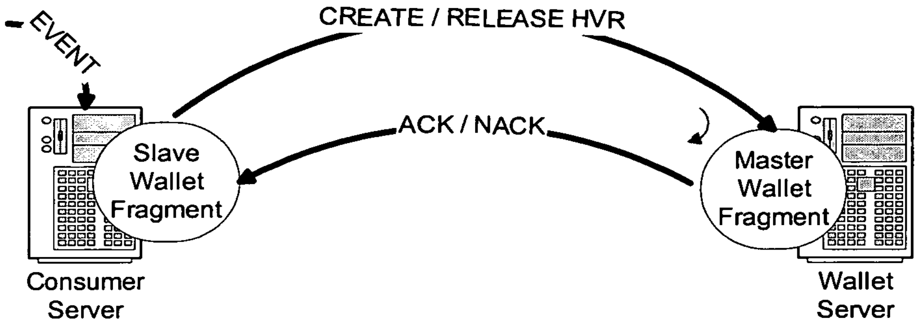


Figure 50K

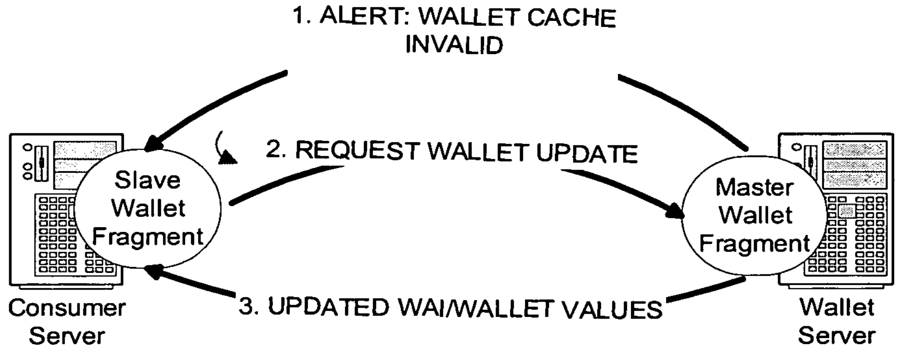


Figure 50L

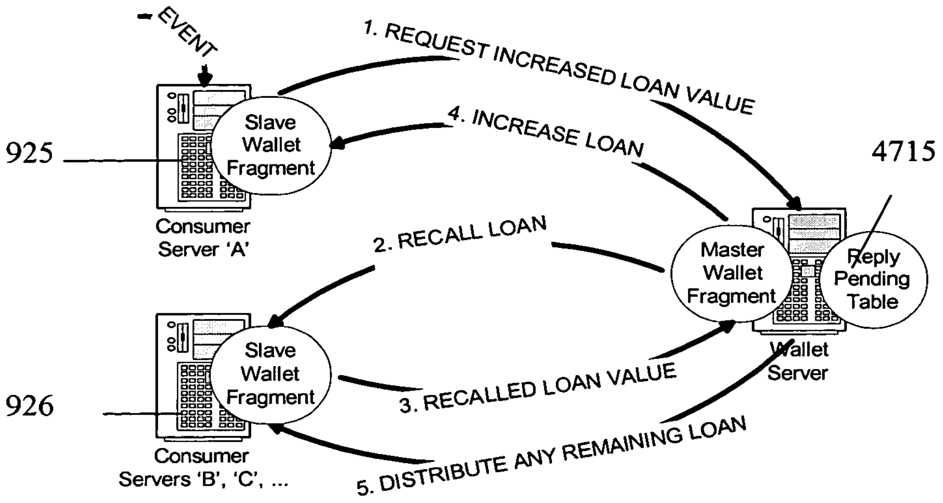

EVENT
1. REQUEST INCREASED LOAN VALUE
4. INCREASE LOAN
925
Slave
Wallet
Fragment
Consumer
Server 'A'
4715
2. RECALL LOAN
Master
Wallet
Fragment
Reply
Pending
Table
Wallet
Server
Slave
Wallet
Fragment
3. RECALLED LOAN VALUE
926
Consumer
Servers 'B', 'C', ...
5. DISTRIBUTE ANY REMAINING LOAN

Figure 51

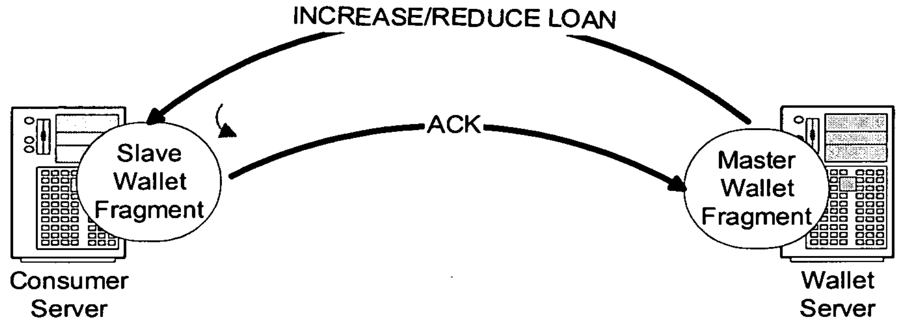


Figure 52

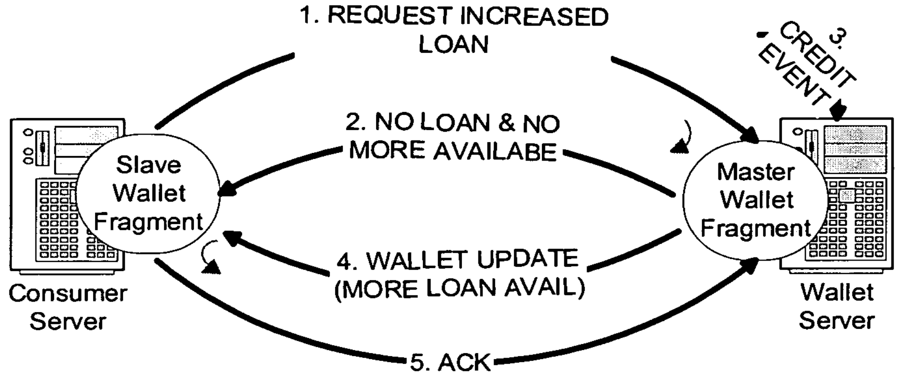


Figure 53

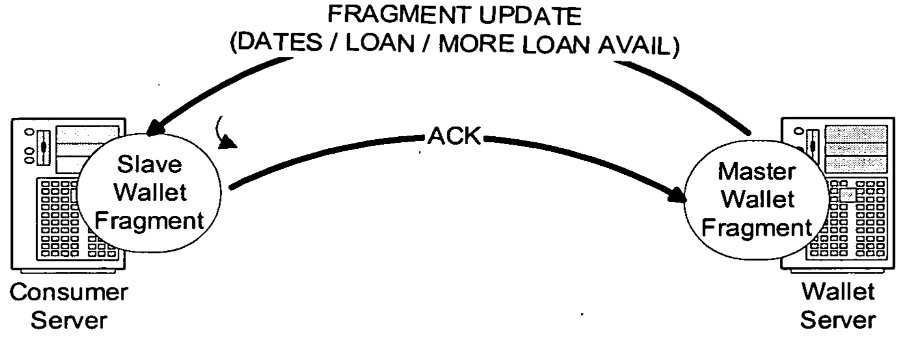


Figure 54

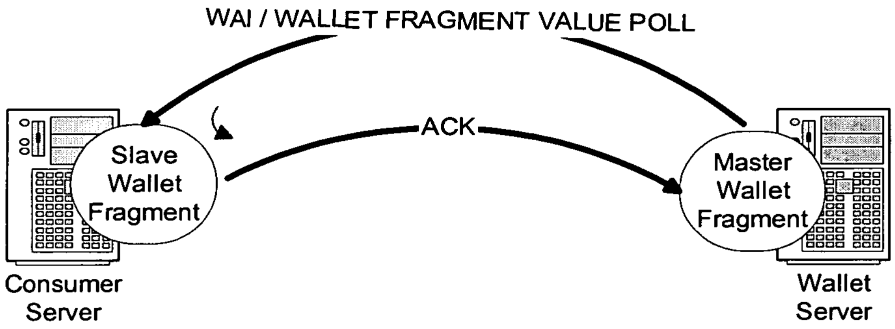


Figure 55

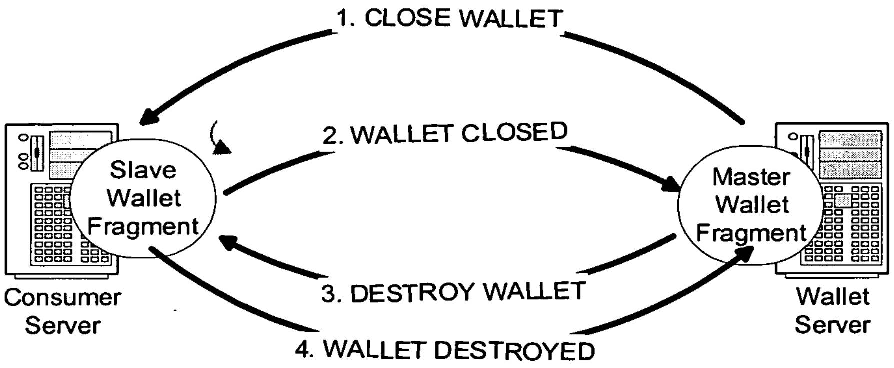


Figure 56

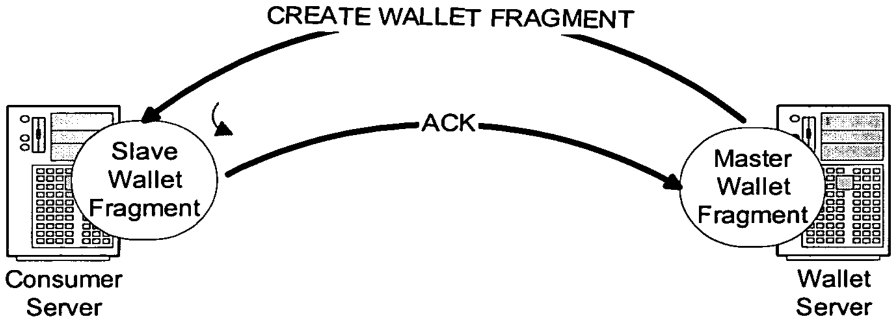
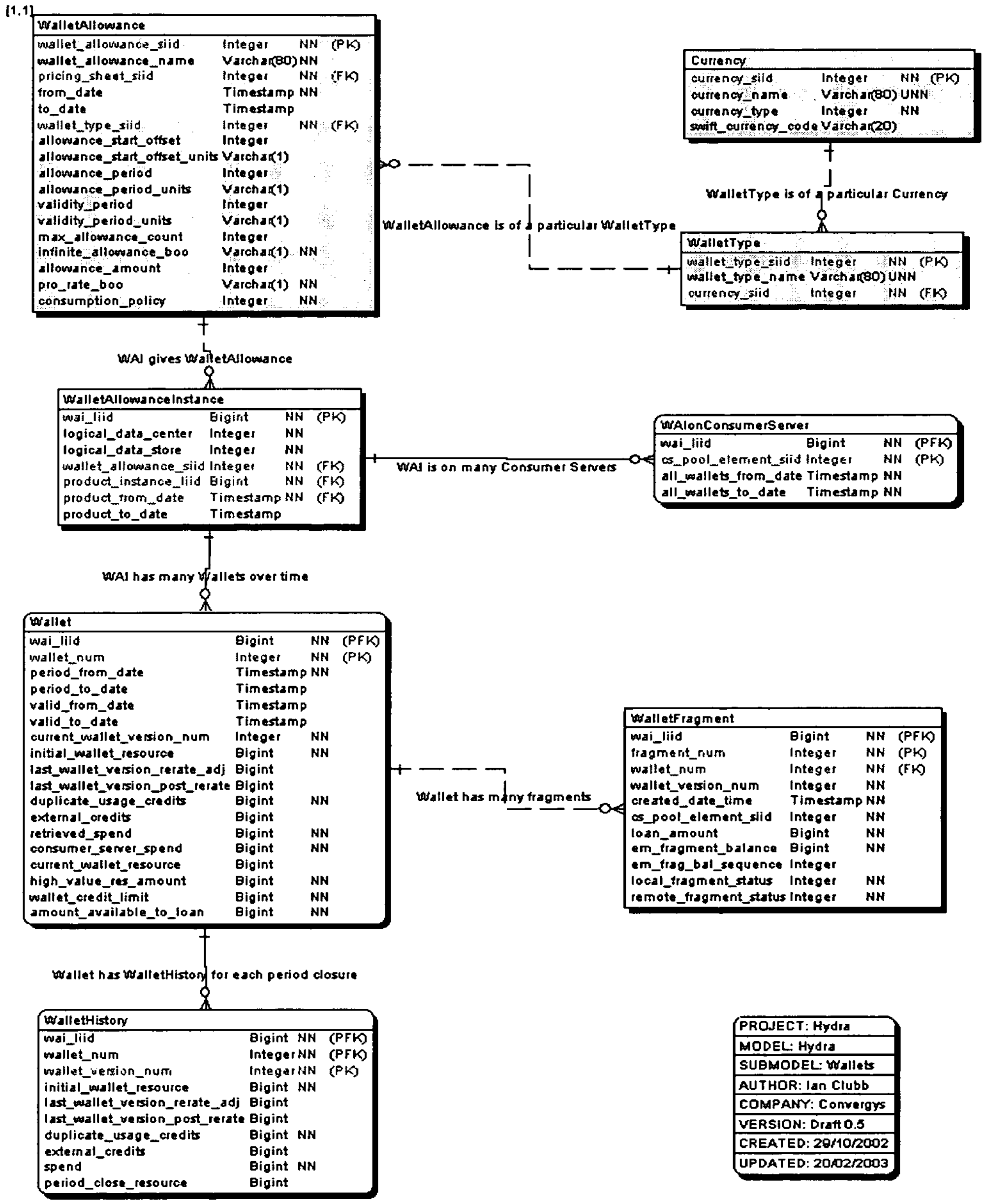


Figure 57


[1,1]
WalletAllowance is of a particular WalletType
WalletType is of a particular Currency
WAI gives WalletAllowance
WAI is on many Consumer Servers
WAI has many Wallets over time
Wallet has many fragments
Wallet has WalletHistory for each period closure

WalletAllowance			
wallet_allowance_siid	Integer	NN	(PK)
wallet_allowance_name	Varchar(80)	NN	
pricing_sheet_siid	Integer	NN	(FK)
from_date	Timestamp	NN	
to_date	Timestamp		
wallet_type_siid	Integer	NN	(FK)
allowance_start_offset	Integer		
allowance_start_offset_units	Varchar(1)		
allowance_period	Integer		
allowance_period_units	Varchar(1)		
validity_period	Integer		
validity_period_units	Varchar(1)		
max_allowance_count	Integer		
infinite_allowance_boo	Varchar(1)	NN	
allowance_amount	Integer		
pro_rate_boo	Varchar(1)	NN	
consumption_policy	Integer	NN	

Currency			
currency_siid	Integer	NN	(PK)
currency_name	Varchar(80)	UNN	
currency_type	Integer	NN	
swift_currency_code	Varchar(20)		

WalletType			
wallet_type_siid	Integer	NN	(PK)
wallet_type_name	Varchar(80)	UNN	
currency_siid	Integer	NN	(FK)

WalletAllowanceInstance			
wai_liid	Bigint	NN	(PK)
logical_data_center	Integer	NN	
logical_data_store	Integer	NN	
wallet_allowance_siid	Integer	NN	(FK)
product_instance_liid	Bigint	NN	(FK)
product_from_date	Timestamp	NN	(FK)
product_to_date	Timestamp		

WAIonConsumerServer			
wai_liid	Bigint	NN	(PFK)
cs_pool_element_siid	Integer	NN	(PK)
all_wallets_from_date	Timestamp	NN	
all_wallets_to_date	Timestamp	NN	

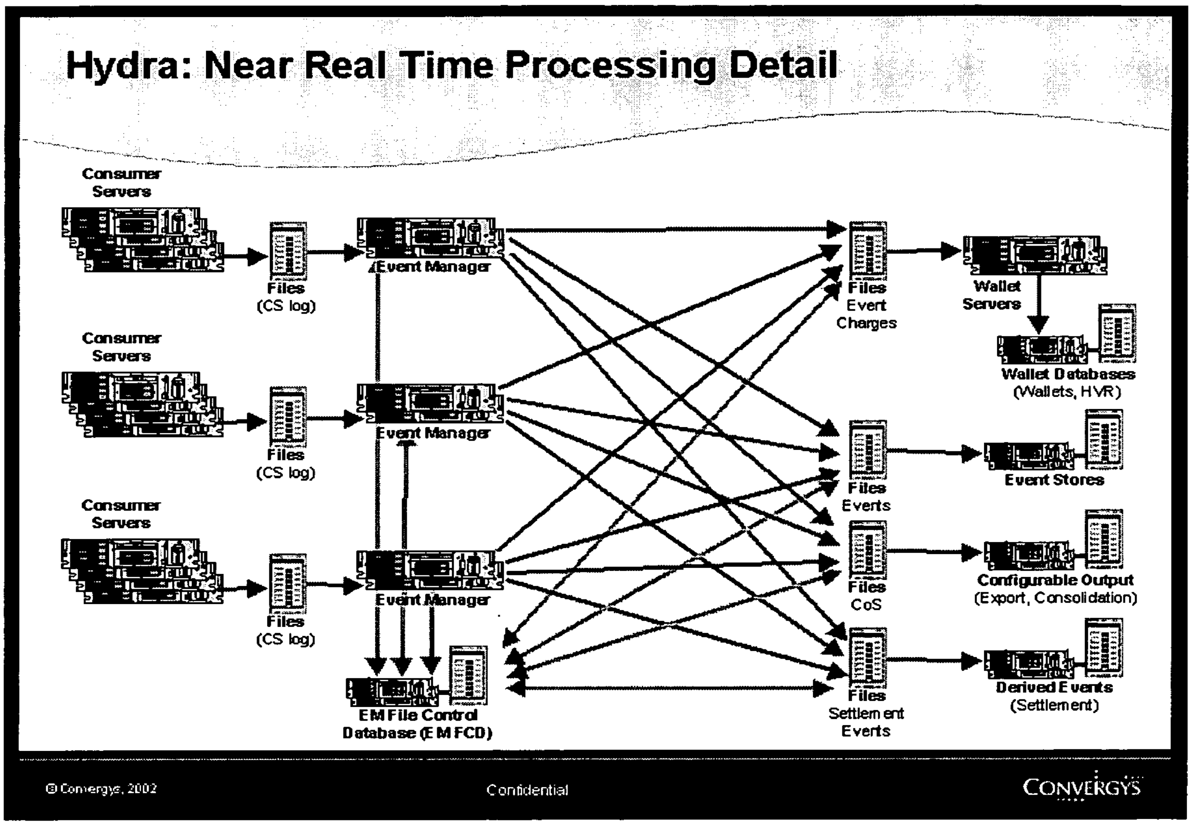
Wallet			
wai_liid	Bigint	NN	(PFK)
wallet_num	Integer	NN	(PK)
period_from_date	Timestamp	NN	
period_to_date	Timestamp		
valid_from_date	Timestamp		
valid_to_date	Timestamp		
current_wallet_version_num	Integer	NN	
initial_wallet_resource	Bigint	NN	
last_wallet_version_rerate_adj	Bigint		
last_wallet_version_post_rerate	Bigint		
duplicate_usage_credits	Bigint	NN	
external_credits	Bigint		
retrieved_spend	Bigint	NN	
consumer_server_spend	Bigint	NN	
current_wallet_resource	Bigint		
high_value_res_amount	Bigint	NN	
wallet_credit_limit	Bigint	NN	
amount_available_to_loan	Bigint	NN	

WalletFragment			
wai_liid	Bigint	NN	(PFK)
fragment_num	Integer	NN	(PK)
wallet_num	Integer	NN	(FK)
wallet_version_num	Integer	NN	
created_date_time	Timestamp	NN	
cs_pool_element_siid	Integer	NN	
loan_amount	Bigint	NN	
em_fragment_balance	Bigint	NN	
em_frag_bal_sequence	Integer		
local_fragment_status	Integer	NN	
remote_fragment_status	Integer	NN	

WalletHistory			
wai_liid	Bigint	NN	(PFK)
wallet_num	Integer	NN	(PFK)
wallet_version_num	Integer	NN	(PK)
initial_wallet_resource	Bigint	NN	
last_wallet_version_rerate_adj	Bigint		
last_wallet_version_post_rerate	Bigint		
duplicate_usage_credits	Bigint	NN	
external_credits	Bigint		
spend	Bigint	NN	
period_close_resource	Bigint		

PROJECT: Hydra
MODEL: Hydra
SUBMODEL: Wallets
AUTHOR: Ian Clubb
COMPANY: Convergys
VERSION: Draft 0.5
CREATED: 29/10/2002
UPDATED: 20/02/2003

Figure 58



Hydra: Near Real Time Processing Detail

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

EM File Control Database (EMFCD)

Files Event Charges

Files Events

Files CoS

Files Settlement Events

Wallet Servers

Wallet Databases (Wallets, HVR)

Event Stores

Configurable Output (Export, Consolidation)

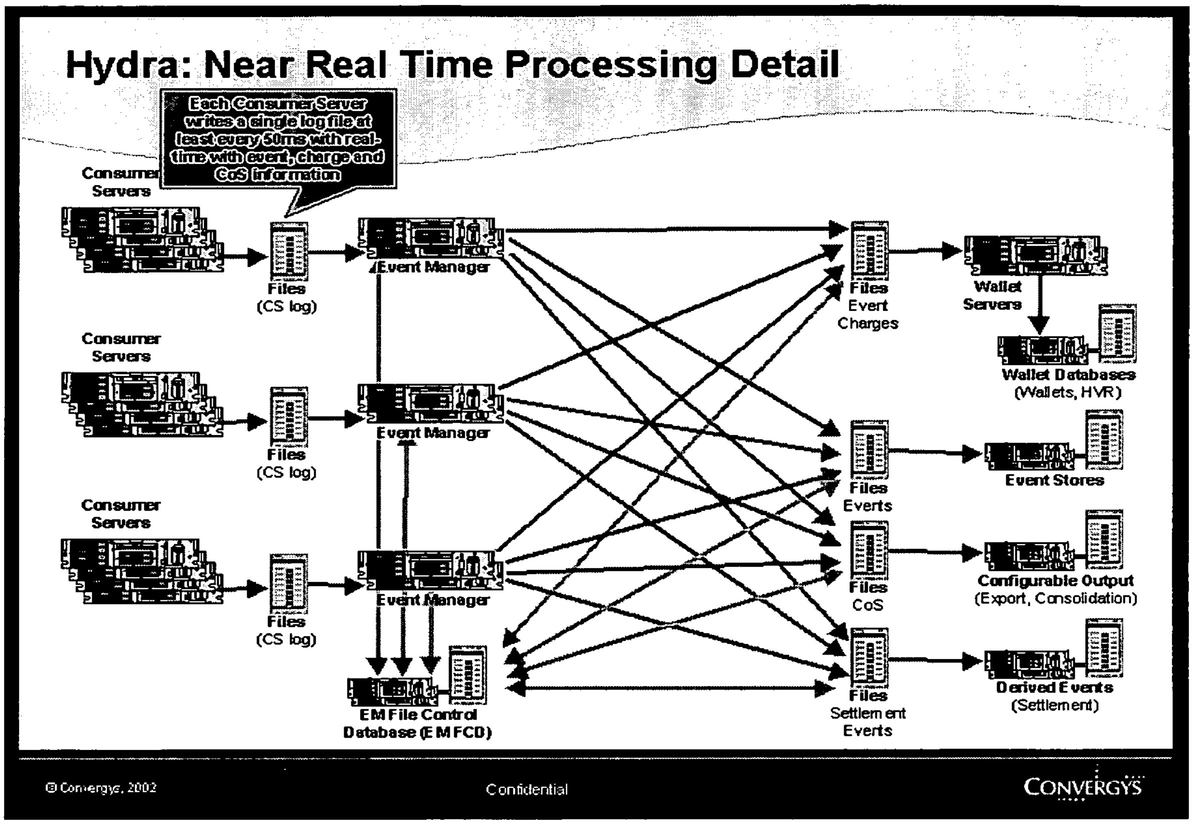
Derived Events (Settlement)

© Convergys, 2002

Confidential

CONVERGYS

Figure 59



Hydra: Near Real Time Processing Detail

Each Consumer Server writes a single log file at least every 50ms with real-time with event, charge and CoS information

Consumer Servers
Files
(CS log)
Event Manager

Consumer Servers
Files
(CS log)
Event Manager

Consumer Servers
Files
(CS log)
Event Manager

EM File Control Database (EMFCD)

Files
Event Charges
Wallet Servers
Wallet Databases
(Wallets, HVR)

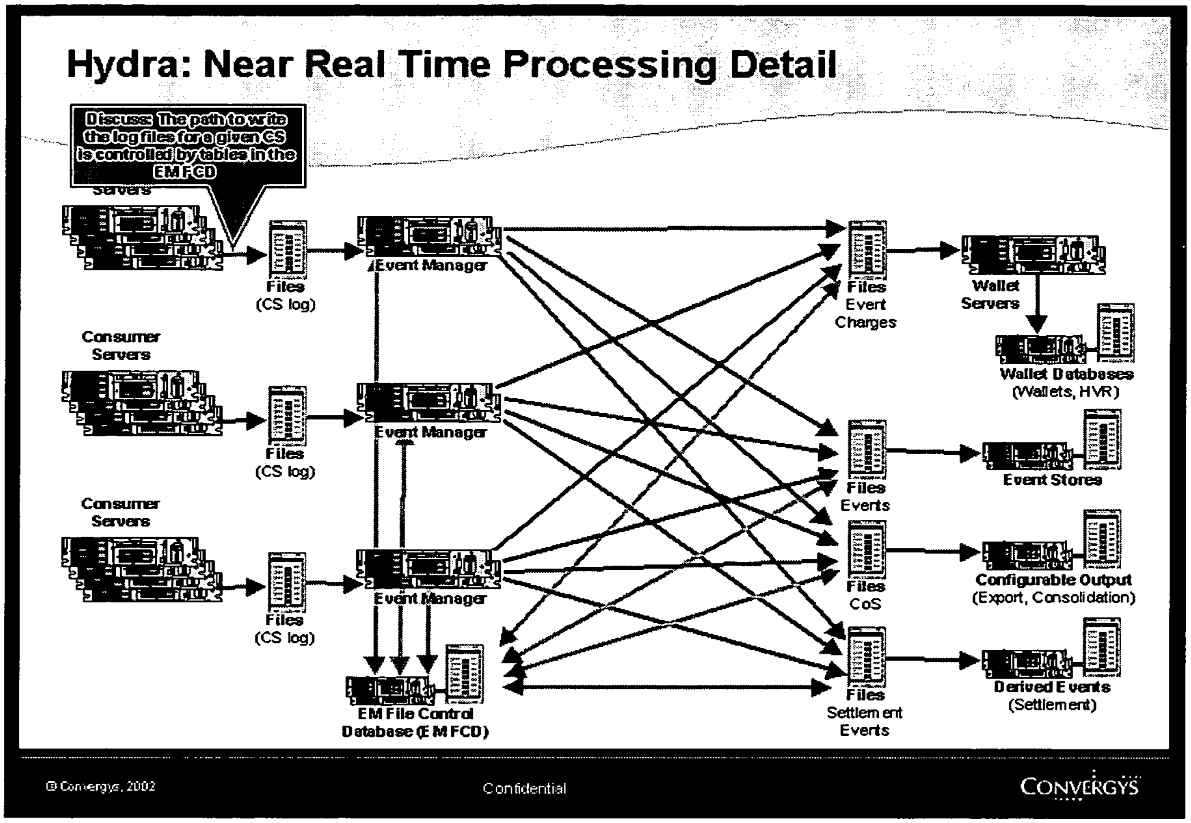
Files
Events
Event Stores

Files
CoS
Configurable Output
(Export, Consolidation)

Files
Settlement Events
Derived Events
(Settlement)

© Convergys, 2002
Confidential
CONVERGYS

Figure 60



Hydra: Near Real Time Processing Detail

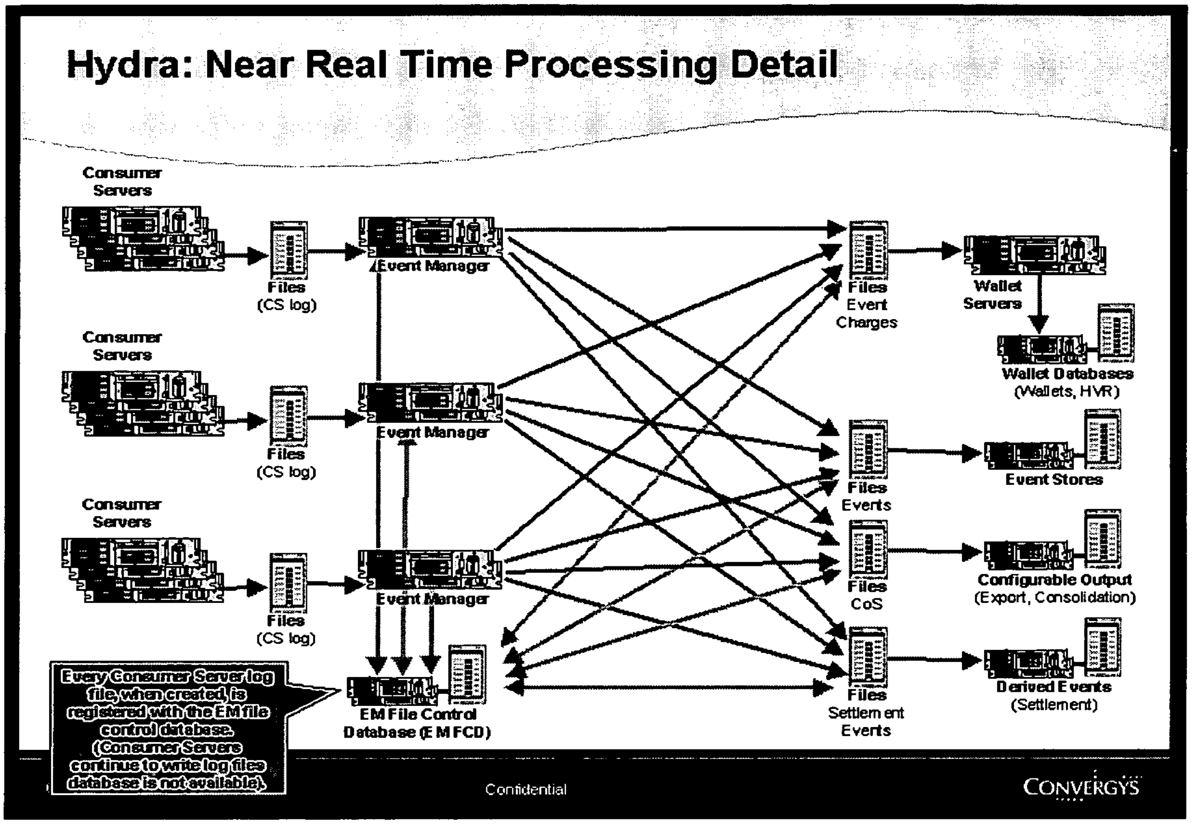
Discuss: The path to write the log files for a given CS is controlled by tables in the EM FCD

© Convergys, 2002

Confidential

CONVERGYS

Figure 61



Hydra: Near Real Time Processing Detail

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Every Consumer Server log file, when created, is registered with the EM file control database. (Consumer Servers continue to write log files database is not available).

EM File Control Database (EMFCD)

Files Event Charges

Wallet Servers

Wallet Databases (Wallets, HVR)

Files Events

Event Stores

Files CoS

Configurable Output (Export, Consolidation)

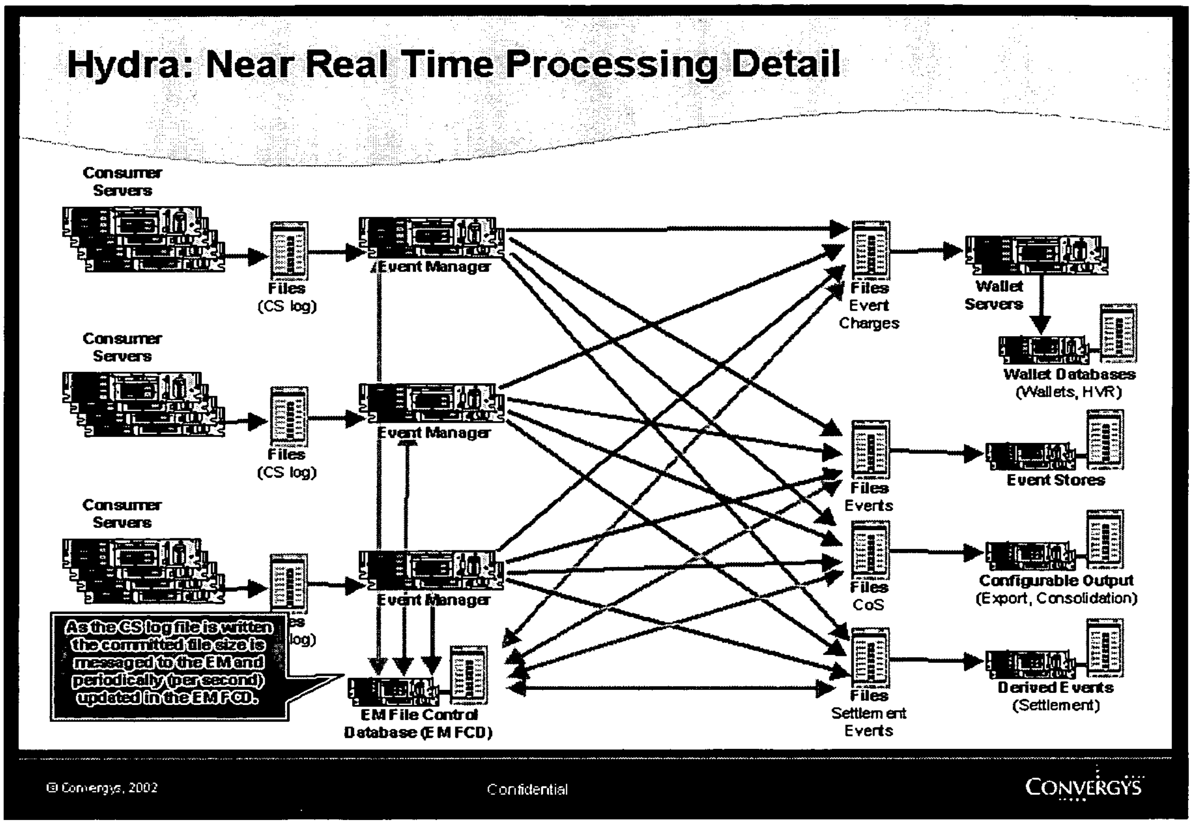
Files Settlement Events

Derived Events (Settlement)

Confidential

CONVERGYS

Figure 62



Hydra: Near Real Time Processing Detail

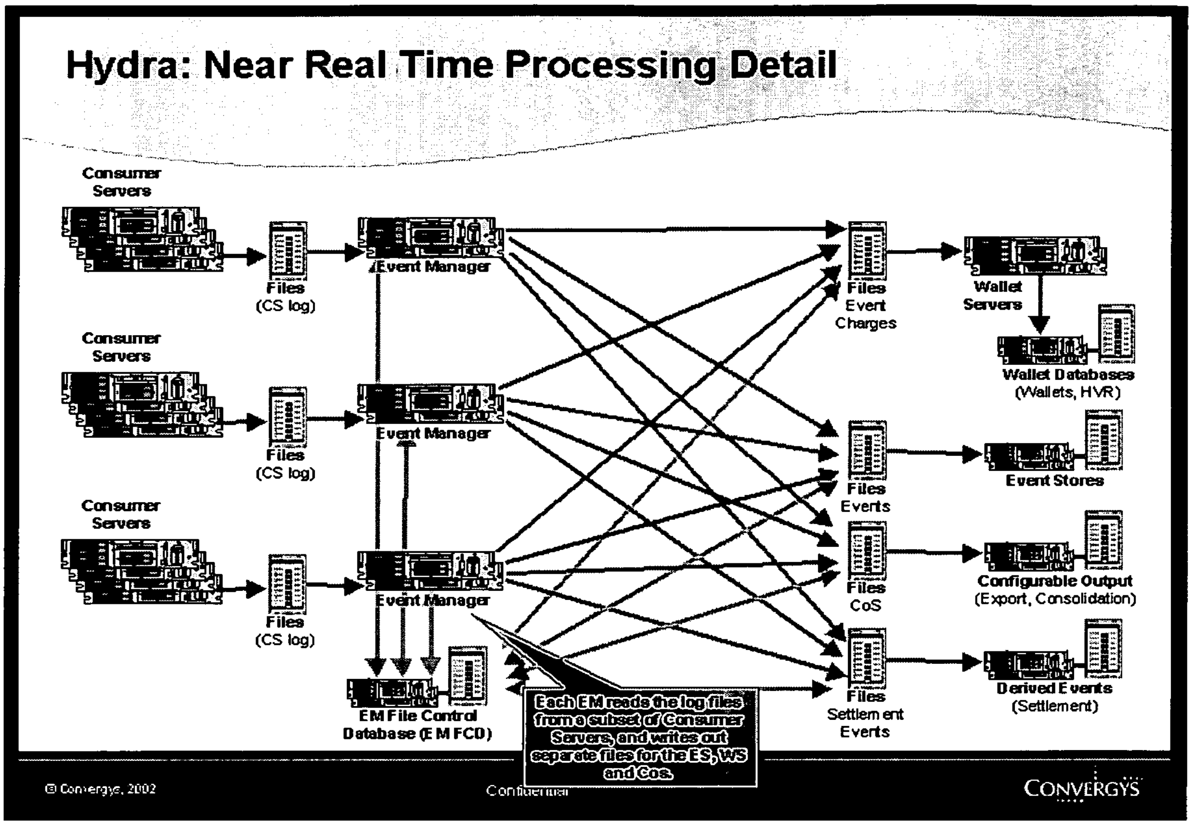
As the CS log file is written the committed file size is messaged to the EM and periodically (per second) updated in the EMFCD.

© Convergys, 2002

Confidential

CONVERGYS

Figure 63



Hydra: Near Real Time Processing Detail

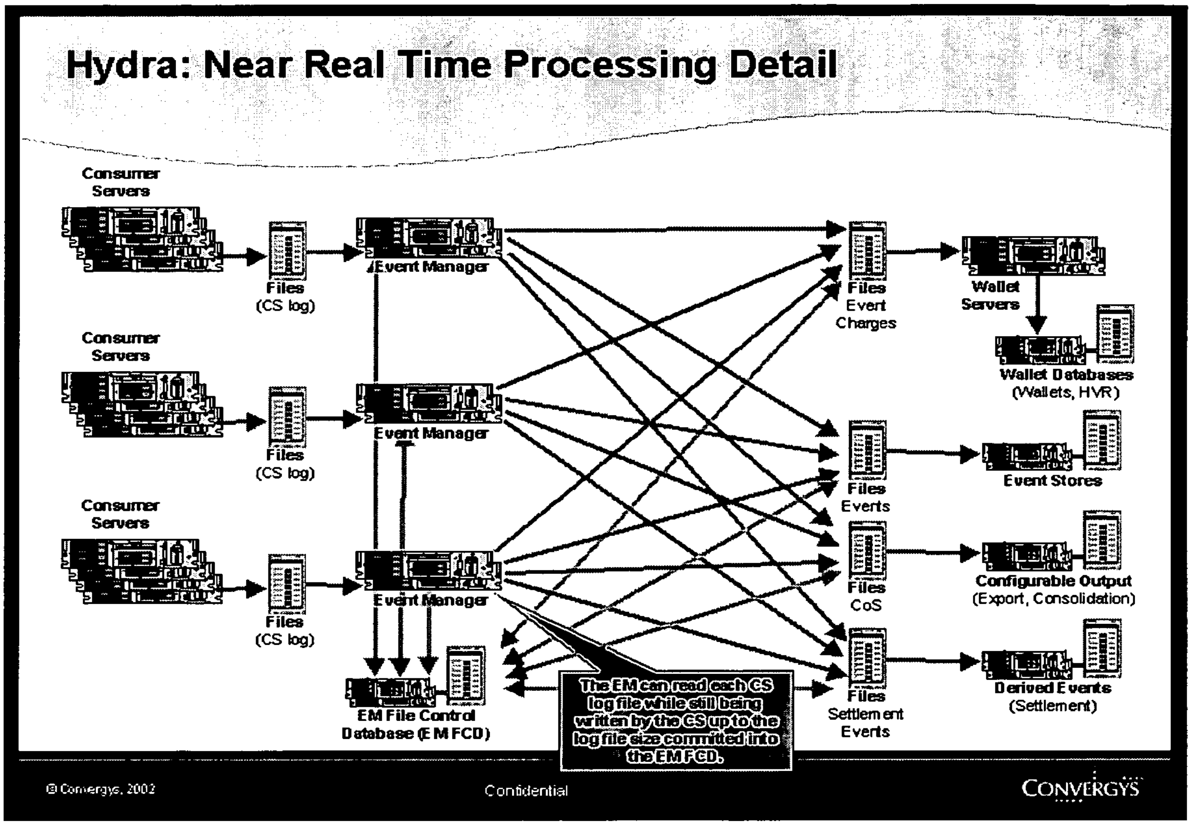
Each EM reads the log files from a subset of Consumer Servers, and writes out separate files for the ES, WS and Cos.

© Convergys, 2002

Confidential

CONVERGYS

Figure 64



Hydra: Near Real Time Processing Detail

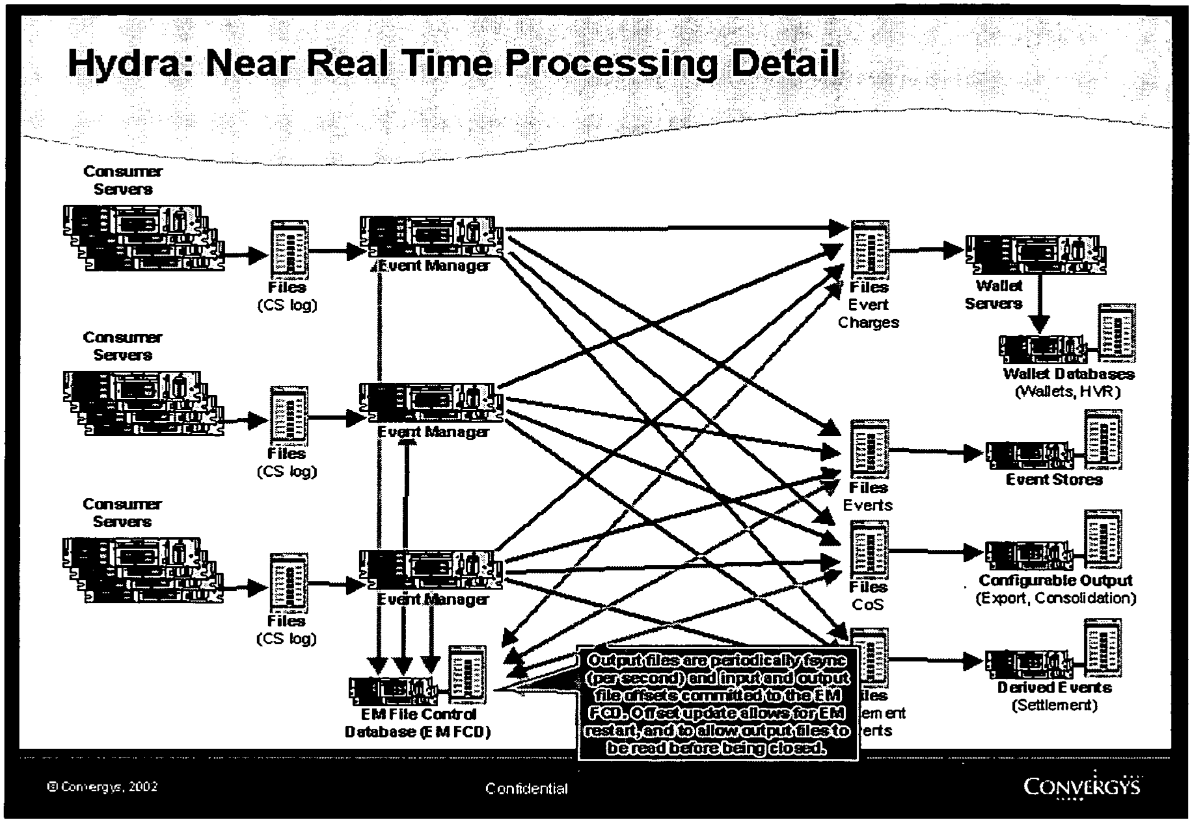
The EM can read each CS log file while still being written by the CS up to the log file size committed into the EM FCD.

© Convergys, 2002

Confidential

CONVERGYS

Figure 65



Hydra: Near Real Time Processing Detail

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

EM File Control Database (EMFCD)

Files Event Charges

Wallet Servers

Wallet Databases (Wallets, HVR)

Files Events

Event Stores

Files CoS

Configurable Output (Export, Consolidation)

iles ement erts

Derived Events (Settlement)

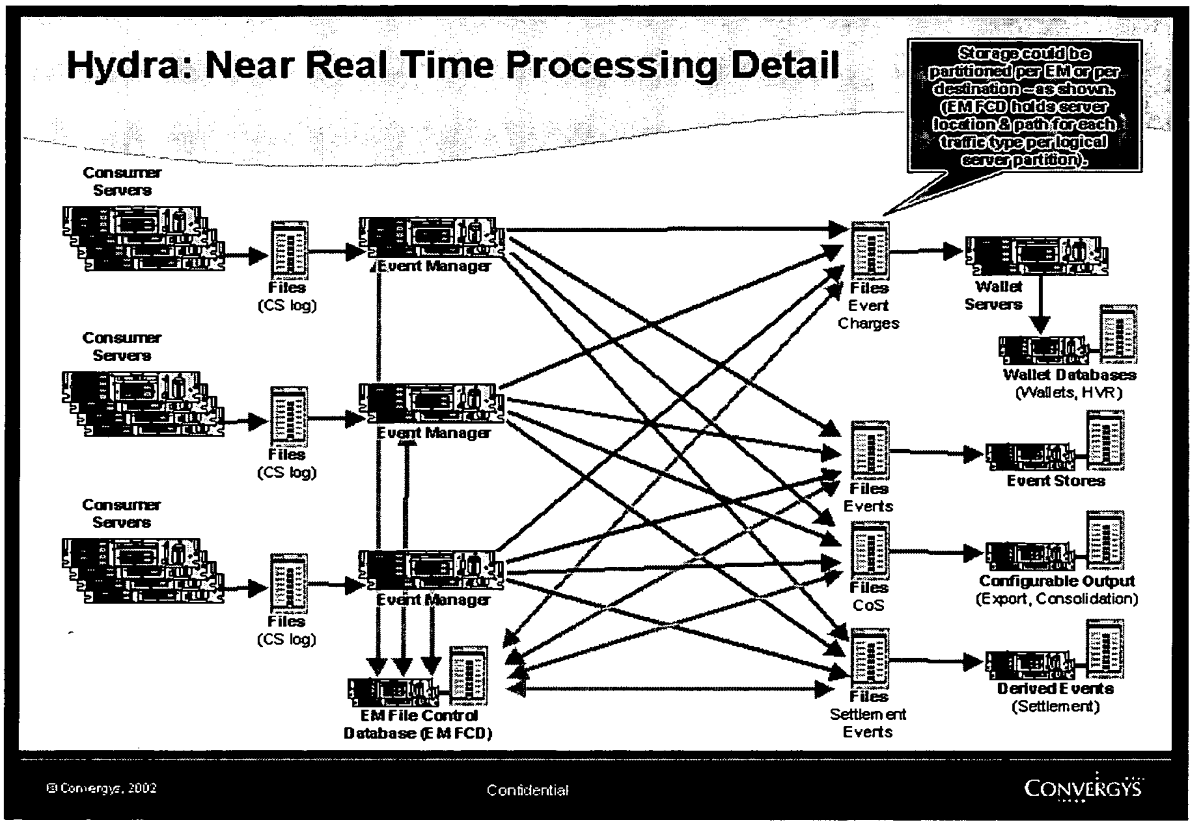
Output files are periodically fsync (per second) and input and output file offsets committed to the EM FCD. Offset update allows for EM restart, and to allow output files to be read before being closed.

© Convergys, 2002

Confidential

CONVERGYS

Figure 66



Hydra: Near Real Time Processing Detail

Storage could be partitioned per EM or per destination – as shown. (EM FCD holds server location & path for each traffic type per logical server partition).

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

EM File Control Database (EM FCD)

Files Event Charges

Wallet Servers

Wallet Databases (Wallets, HVR)

Files Events

Event Stores

Files CoS

Configurable Output (Export, Consolidation)

Files Settlement Events

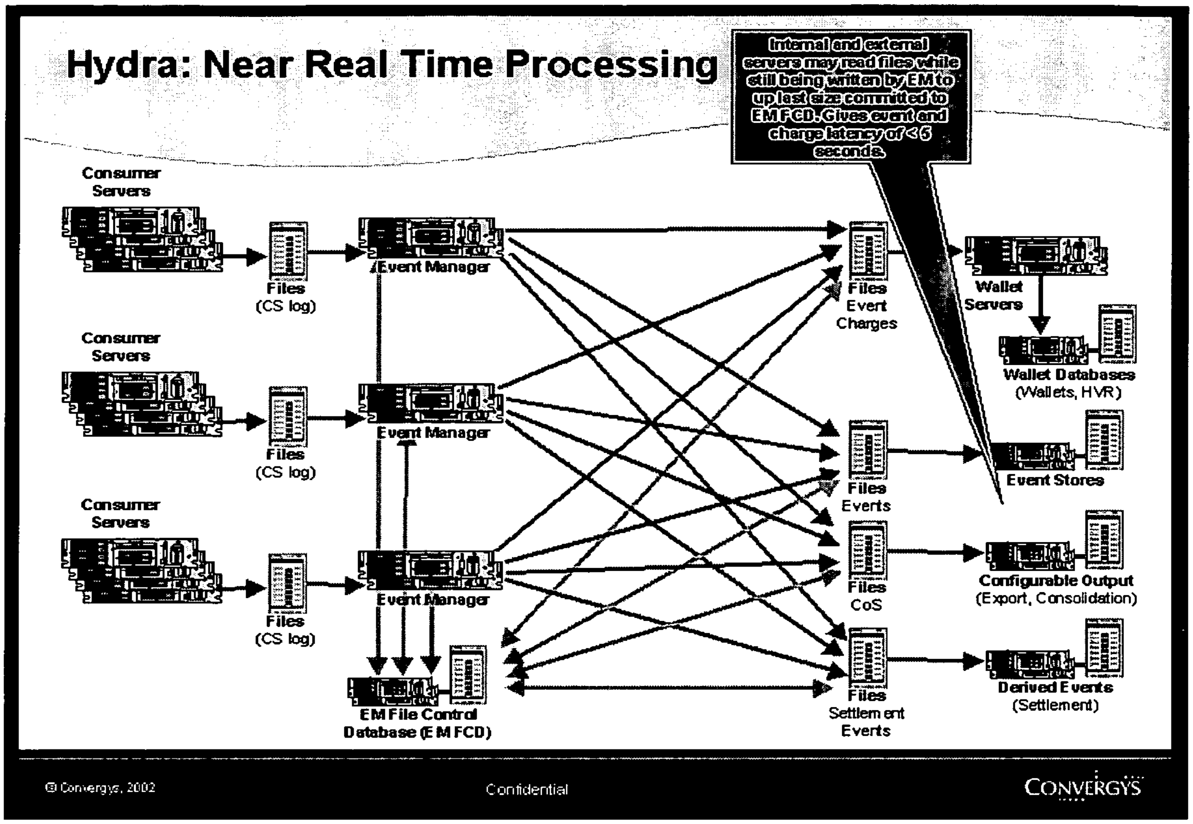
Derived Events (Settlement)

© Convergys, 2002

Confidential

CONVERGYS

Figure 67



Hydra: Near Real Time Processing

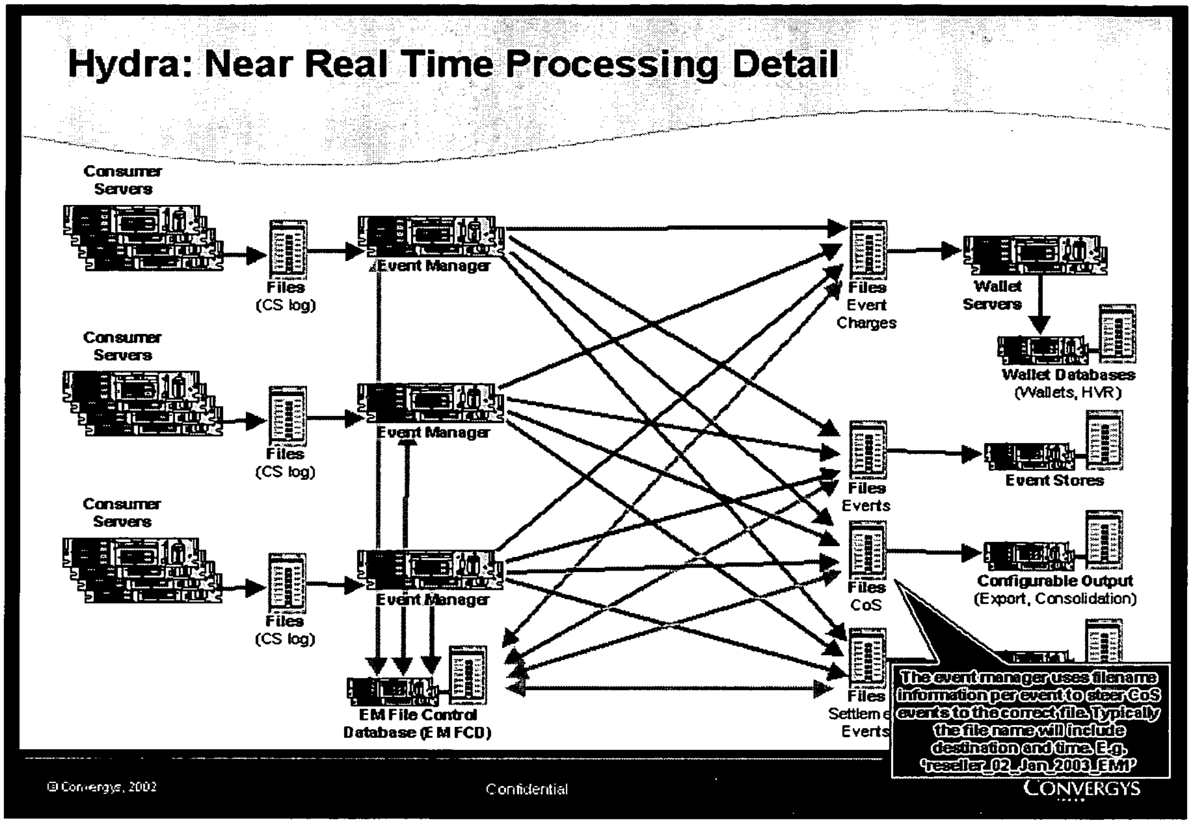
Internal and external servers may read files while still being written by EM to up last size committed to EM FCD. Gives event and charge latency of < 5 seconds.

© Convergys, 2002

Confidential

CONVERGYS

Figure 68



Hydra: Near Real Time Processing Detail

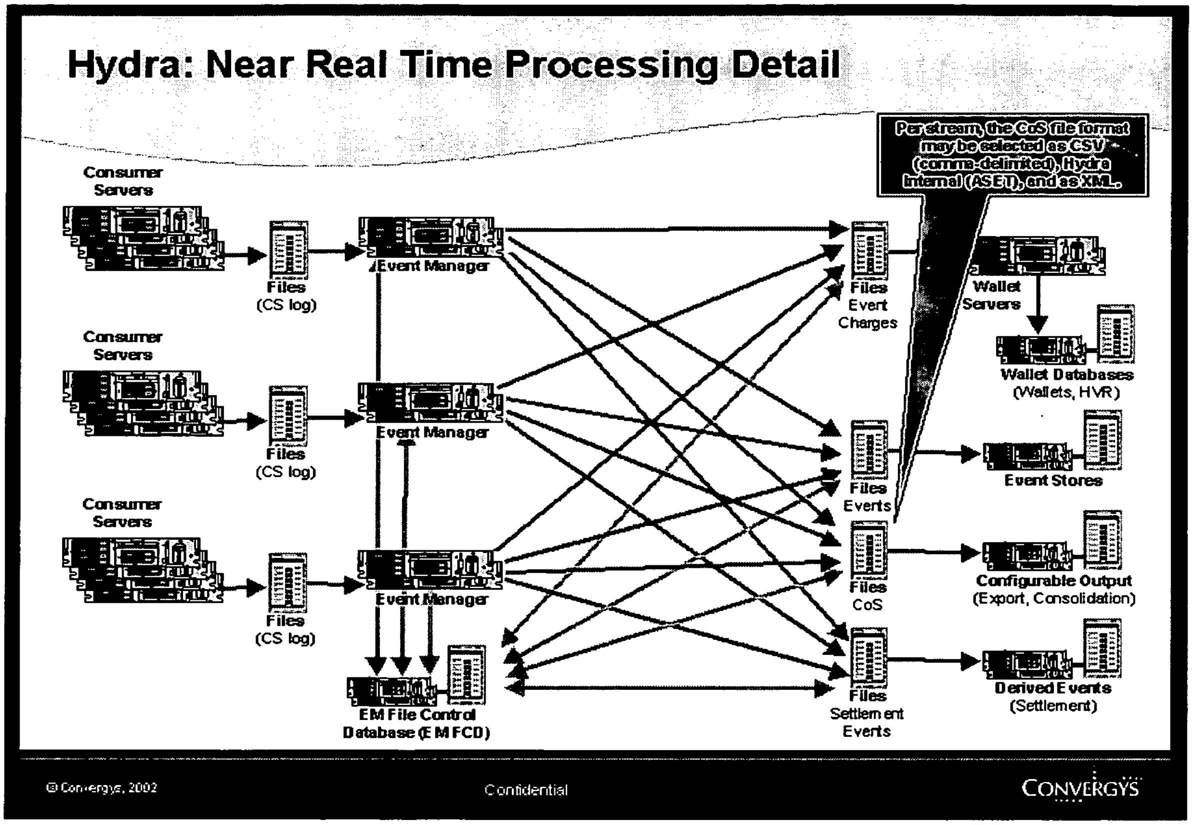
The event manager uses filename information per event to steer CoS events to the correct file. Typically the file name will include destination and time. E.g. 'reseller_02_Jan_2003_EMI'

© Convergys, 2002

Confidential

CONVERGYS

Figure 69



Hydra: Near Real Time Processing Detail

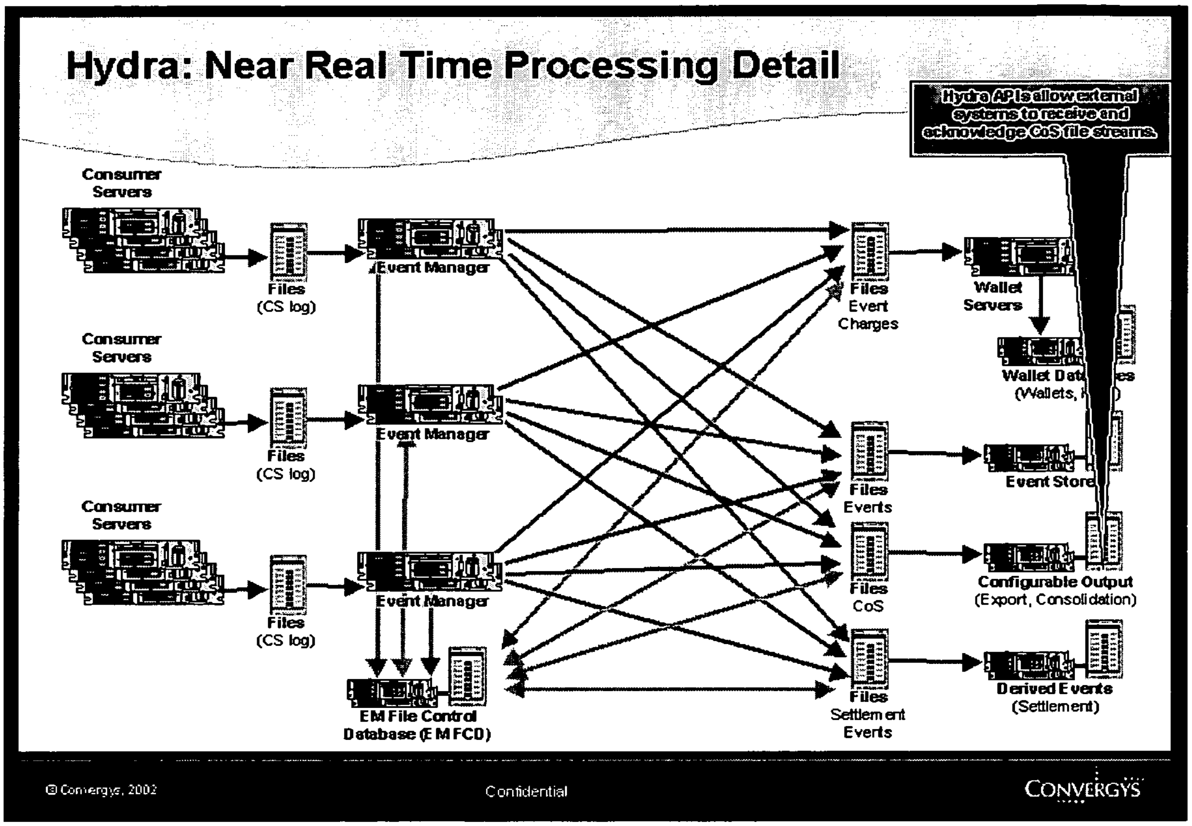
Per stream, the CoS file format may be selected as CSV (comma-delimited), Hydra Internal (ASET), and as XML.

© Convergys, 2002

Confidential

CONVERGYS

Figure 70



Hydra: Near Real Time Processing Detail

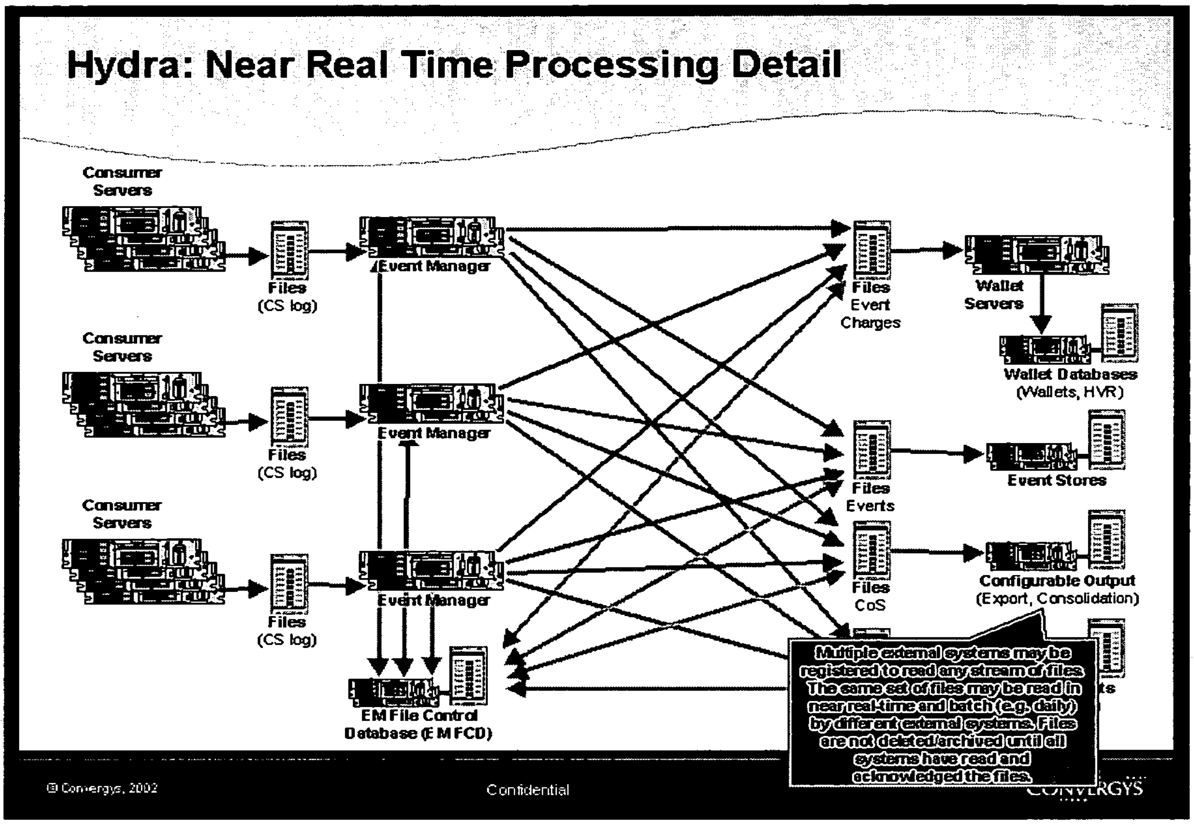
Hydra APIs allow external systems to receive and acknowledge CoS file streams.

© Convergys, 2002

Confidential

CONVERGYS

Figure 71



Hydra: Near Real Time Processing Detail

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

EM File Control Database (EMFCD)

Files Event Charges

Wallet Servers

Wallet Databases (Wallets, HVR)

Files Events

Event Stores

Files CoS

Configurable Output (Export, Consolidation)

ts

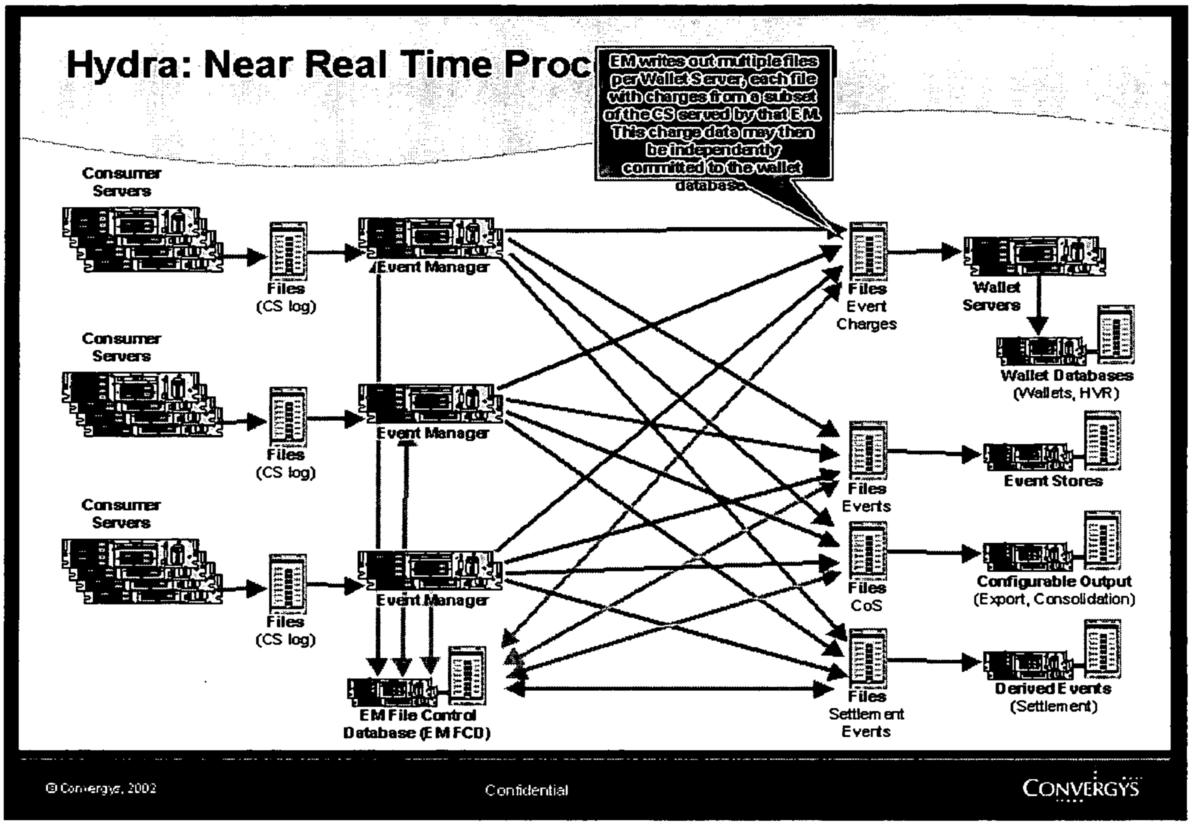
Multiple external systems may be registered to read any stream of files. The same set of files may be read in near real-time and batch (e.g. daily) by different external systems. Files are not deleted/archived until all systems have read and acknowledged the files.

© Convergys, 2002

Confidential

CONVERGYS

Figure 72



Hydra: Near Real Time Proc

EM writes out multiple files per Wallet Server, each file with charges from a subset of the CS served by that EM. This charge data may then be independently committed to the wallet database.

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

Consumer Servers

Files (CS log)

Event Manager

EM File Control Database (EMFCD)

Files Event Charges

Wallet Servers

Wallet Databases (Wallets, HVR)

Files Events

Event Stores

Files CoS

Configurable Output (Export, Consolidation)

Files Settlement Events

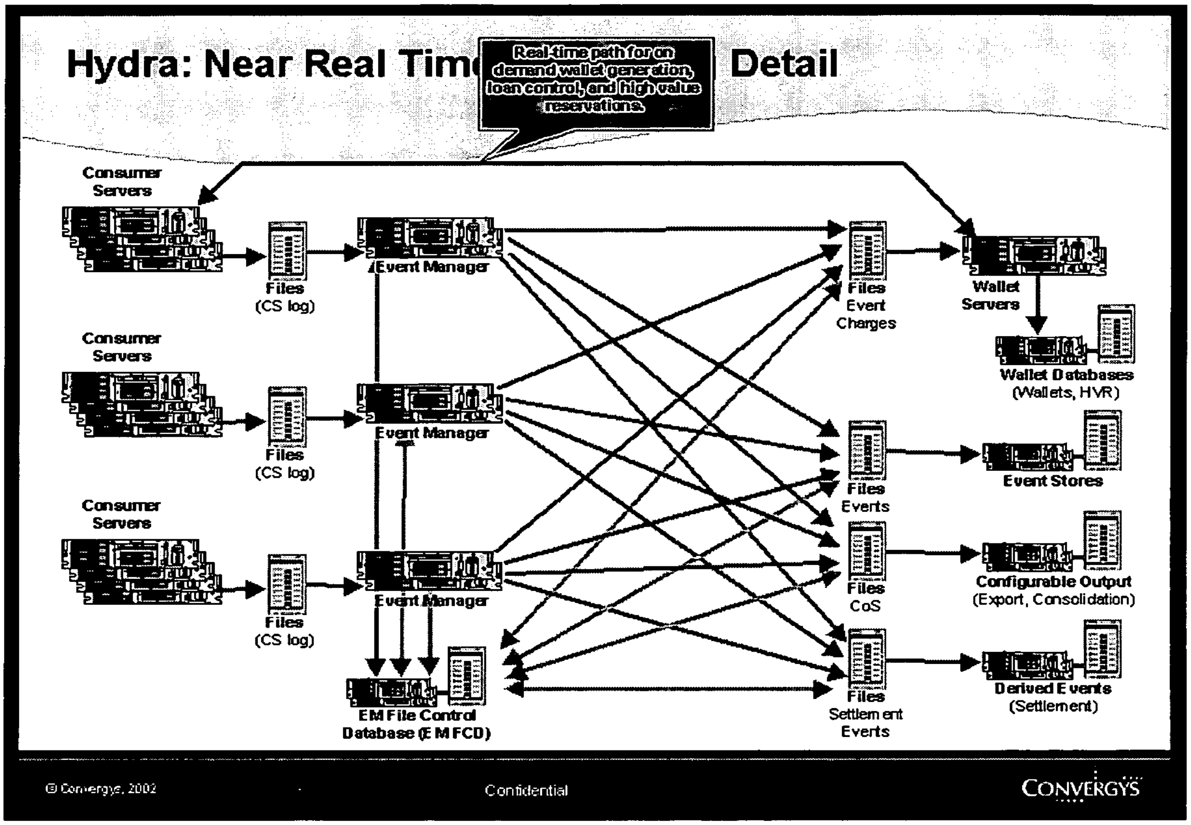
Derived Events (Settlement)

© Convergys, 2002

Confidential

CONVERGYS

Figure 73

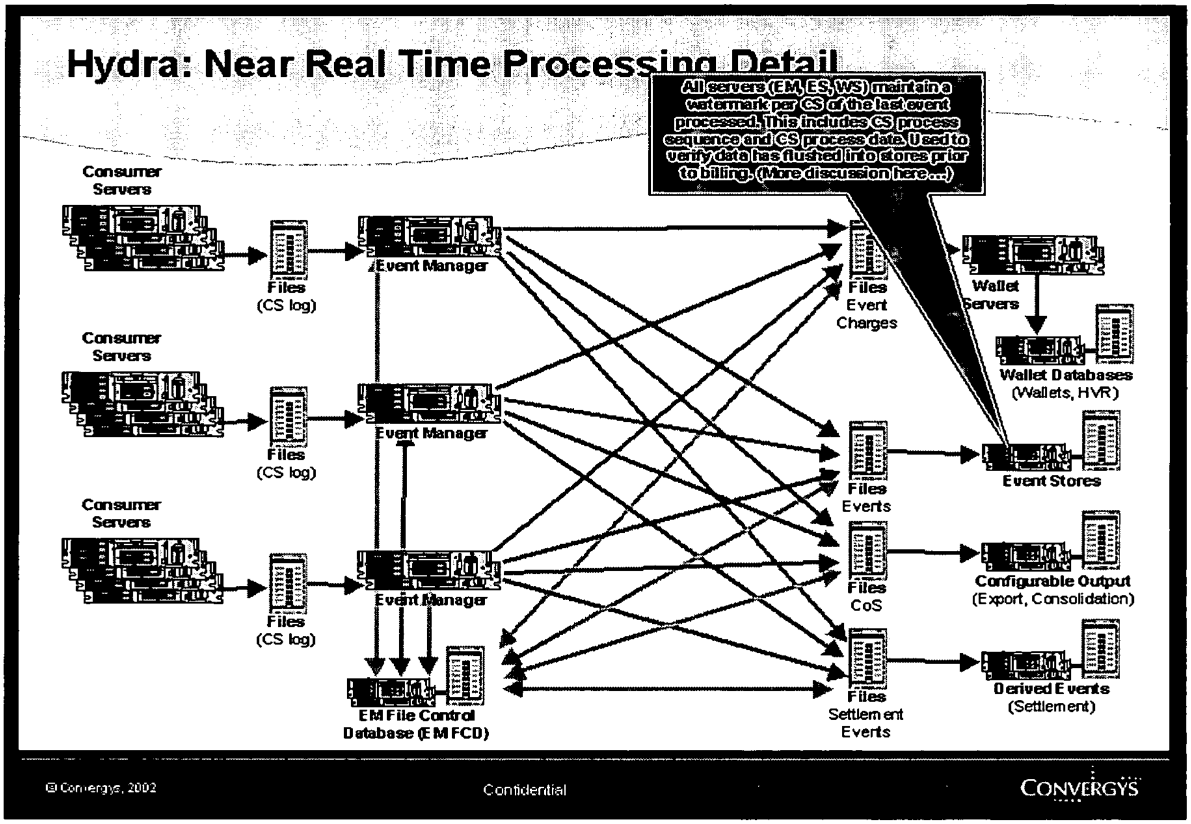


Hydra: Near Real Tim Detail

Real-time path for on demand wallet generation, loan control, and high value reservations.

© Convergys, 2002 Confidential CONVERGYS

Figure 74



Hydra: Near Real Time Processing Detail

All servers (EM, ES, WS) maintain a watermark per CS of the last event processed. This includes CS process sequence and CS process date. Used to verify data has flushed into stores prior to billing. (More discussion here...)

Consumer Servers

Files
(CS log)

Event Manager

Consumer Servers

Files
(CS log)

Event Manager

Consumer Servers

Files
(CS log)

Event Manager

EM File Control Database (EMFCD)

Files
Event Charges

Files
Events

Files
CoS

Files
Settlement Events

Wallet Servers

Wallet Databases
(Wallets, HVR)

Event Stores

Configurable Output
(Export, Consolidation)

Derived Events
(Settlement)

© Convergys, 2002

Confidential

CONVERGYS

Figure 75